

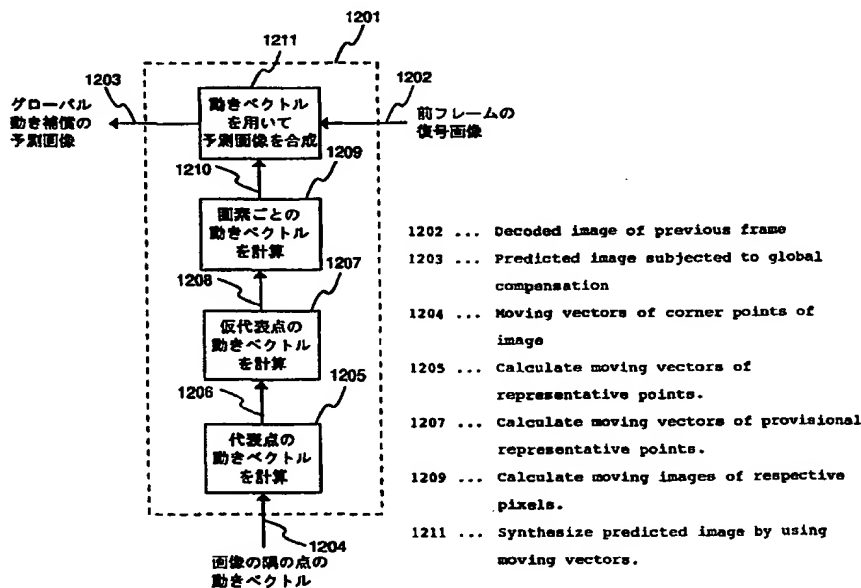
(51) 国際特許分類6 H04N 7/32	A1	(11) 国際公開番号 WO98/56185 (43) 国際公開日 1998年12月10日(10.12.98)
<p>(21) 国際出願番号 PCT/JP98/02435</p> <p>(22) 国際出願日 1998年6月2日(02.06.98)</p> <p>(30) 優先権データ 特願平9/144916 1997年6月3日(03.06.97) JP</p> <p>(71) 出願人 (米国を除くすべての指定国について) 株式会社 日立製作所(HITACHI, LTD.)(JP/JP) 〒101-8010 東京都千代田区神田駿河台四丁目6番地 Tokyo, (JP)</p> <p>(72) 発明者 ; および</p> <p>(75) 発明者 / 出願人 (米国についてのみ) 中屋雄一郎(NAKAYA, Yuichiro)(JP/JP) 〒185-8601 東京都国分寺市東恋ヶ窪一丁目280番地 株式会社 日立製作所 中央研究所内 Tokyo, (JP)</p> <p>(74) 代理人 弁理士 高橋明夫(TAKAHASHI, Akio) 〒103-0025 東京都中央区日本橋茅場町二丁目9番8号 友泉茅場町ビル 日東国際特許事務所 Tokyo, (JP)</p>	<p>(81) 指定国 CN, JP, KR, SG, US, 欧州特許 (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).</p> <p>添付公開書類 国際調査報告書</p>	

(54)Title: IMAGE ENCODING AND DECODING METHOD AND DEVICE

(54)発明の名称 画像符号化及び復号化方法及び装置

(57) Abstract

A method for simplifying the arithmetic operation of the global motion compensation and warping prediction in encoding and decoding the image signal of motion compensation. In a process for synthesizing a predicted image (1203) subjected to global motion compensation and formed to predict the present frame image from a previous frame image (1202) by using moving vectors (1205) of representative points which are spaced at spacial intervals having specific features in an image frame, the moving vectors of provisional representative points are determined from the moving vectors (1206) of the representative points by first interpolation/extrapolation processes (1207), and then the moving vectors (1210) of the pixels are determined from the moving vectors (1208) of the provisional representative points by second interpolation/extrapolation processes (1209). Since the division for synthesizing the predicted image is substituted by a shift calculation with a small number of bits shifted, the processing with a computer or an exclusive hardware can be simplified.



(57)要約

動き補償の画像信号の符号化及び複合におけるグローバル動き補償及びワーピング予測の処理の演算処理を簡略化するため方法を提供する。

画像フレームの空間的な間隔が特別な特徴をもつ複数の代表点の動きベクトル1205を用いて、前フレーム画像1202から現フレーム画像を予測したグローバル動き補償した予測画像1203を合成する処理において、代表点の動きベクトル1206から仮代表点の動きベクトルを求める第1の内・外挿処理1207を行い、その後、仮代表点の動きベクトル1208から画素ごとの動きベクトル1210を求める第2の内・外挿処理1209を行う。

予測画像を合成する際の除算がシフトされるビット数が少ないシフト演算によって代用できるため、計算機又は専用ハードウェアによる処理を簡略化することができる。

PCTに基づいて公開される国際出願のパンフレット第一頁に掲載されたPCT加盟国を同定するために使用されるコード(参考情報)

AL	アルバニア	FI	フィンランド	LK	スリ・ランカ	SI	スロヴェニア
AM	アルメニア	FR	フランス	LR	リベリア	SK	スロヴァキア
AT	オーストリア	GA	ガボン	LS	レソト	SL	シエラ・レオネ
AU	オーストラリア	GB	英国	LT	リトアニア	SN	セネガル
AZ	アゼルバイジャン	GD	グレナダ	LU	ルクセンブルグ	SZ	スワジランド
BA	ボスニア・ヘルツェゴビナ	GE	グルジア	LV	ラトヴィア	TD	チャード
BB	バルバドス	GH	ガーナ	MC	モナコ	TG	トーゴ
BE	ベルギー	GM	ガンビア	MD	モルドヴァ	TJ	タジキスタン
BF	ブルキナ・ファソ	GN	ギニア	MG	マダガスカル	TM	トルクメニスタン
BG	ブルガリア	GW	ギニア・ビサウ	MK	マケドニア旧ユーゴスラヴィア共和国	TR	トルコ
BJ	ベナン	GR	ギリシャ	ML	マリ	TT	トリニダード・トバゴ
BR	ブラジル	HR	クロアチア	MN	モンゴル	UG	ウガンダ
BY	ベラルーシ	HU	ハンガリー	MR	モーリタニア	UA	ウクライナ
CA	カナダ	ID	インドネシア	MW	マラウイ	US	米国
CF	中央アフリカ	IE	アイルランド	MX	メキシコ	UZ	ウズベキスタン
CG	コンゴ	IL	イスラエル	NE	ニジェール	VN	ヴェトナム
CH	スイス	IN	インド	NL	オランダ	YU	ユーゴスラビア
CI	コートジボアール	IS	アイスランド	NO	ノールウェー	ZW	ジンバブエ
CM	カメルーン	IT	イタリア	NZ	ニュージーランド		
CN	中国	JP	日本	PL	ポーランド		
CU	キューバ	KE	ケニア	PT	ポルトガル		
CY	キプロス	KG	キルギスタン	RO	ルーマニア		
CZ	チェッコ	KP	北朝鮮	RU	ロシア		
DE	ドイツ	KR	韓国	SD	スーダン		
DK	デンマーク	KZ	カザフスタン	SE	スウェーデン		
EE	エストニア	LC	セントルシア	SG	シンガポール		
ES	スペイン	LI	リヒテンシュタイン				

明細書

画像符号化及び復号化方法及び装置

技術分野

5 本発明は、画像符号化及び復号化方法及び装置、さらに詳しく言えば、動画像を含む画像信号の符号化、復号化の処理において、代表点の動きベクトルに対して内・外挿処理を行うことにより画像内の画素の動きベクトルを計算するフレーム間予測画像の合成処理を含む画像符号化及び復号化方法及び装置に関するものである。

10 発明の背景

15 動画像の高能率符号化において、異なる時間に発生したフレーム間の類似性を活用するフレーム間予測（動き補償）は情報圧縮に大きな効果を示すことが知られている。現在の画像符号化技術の主流となっている動き補償方式は、動画像符号化方式の国際標準であるH. 261、MPEG1、MPEG2に採用されているブロックマッチング方式である。この方式では、符号化しようとする画像を多数のブロックに分割し、ブロックごとにその動きベクトルを求める。

ブロックマッチングは現在最も広く利用されている動き補償方式であるが、画像全体が拡大・縮小・回転している場合には、全てのブロック
20 に対して動きベクトルを伝送しなければならず、符号化効率が悪くなる
問題がある。この問題に対し、画像全体の動きベクトル場を少ないパラ
メータを用いて表現するグローバル動き補償（例えば、M.Hotter,
"Differential estimation of the global motion parameters zoom and
pan", Signal Processing, vol. 16, no. 3, pp. 249-265, Mar. 1989）が提
25 案されている。これは、画像内の画素（ x , y ）の動きベクトル（ u_g
（ x , y ）, v_g （ x , y ））を、

$$\begin{aligned} u_g(x, y) &= a_0x + a_1y + a_2 \\ v_g(x, y) &= a_3x + a_4y + a_5 \end{aligned} \quad \dots (1)$$

や、

$$\begin{aligned} u_g(x, y) &= b_0xy + b_1x + b_2y + b_3 \\ v_g(x, y) &= b_4xy + b_5x + b_6y + b_7 \end{aligned} \quad \dots (2)$$

5

の形式で表し、この動きベクトルを利用して動き補償を行う方式である。
 ここで $a_0 \sim a_5$ 、 $b_0 \sim b_7$ は動きパラメータである。動き補償を行う
 際には、送信側と受信側で同じ予測画像が得られなければならない。こ
 のために、送信側は受信側へ $a_0 \sim a_5$ 又は $b_0 \sim b_7$ の値を直接伝送
 10 しても良いが、代わりにいくつかの代表点の動きベクトルを伝送する方
 法もある。いま、画像の左上端、右上端、左下端、右下端の画素の座標
 がそれぞれ $(0, 0)$ 、 $(r, 0)$ 、 $(0, s)$ 、 (r, s) で表され
 るとする（ただし、 r と s は正の整数）。このとき、代表点 $(0, 0)$ 、
 $(r, 0)$ 、 $(0, s)$ の動きベクトルの水平・垂直成分をそれぞれ $(u$
 15 $a, v_a)$ 、 (u_b, v_b) 、 (u_c, v_c) とすると、式(1)は

$$\begin{aligned} u_g(x, y) &= \frac{u_b - u_a}{r} x + \frac{u_c - u_a}{s} y + u_a \\ v_g(x, y) &= \frac{v_b - v_a}{r} x + \frac{v_c - v_a}{s} y + v_a \end{aligned} \quad \dots (3)$$

となる。このことは $a_0 \sim a_5$ を伝送する代わりに u_a, v_a, u_b, v
 b, u_c, v_c を伝送しても同様の機能が実現できることを意味する。こ
 20 れと同じように、4個の代表点 $(0, 0)$ 、 $(r, 0)$ 、 $(0, s)$ 、
 (r, s) の動きベクトルの水平・垂直成分 (u_a, v_a) 、 $(u_b, v$
 $b)$ 、 (u_c, v_c) 、 (u_d, v_d) を用いて式(2)は、

$$\begin{aligned} u_g(x, y) &= \frac{s-y}{s} \left(\frac{r-x}{r} u_a + \frac{x}{r} u_b \right) + \frac{y}{s} \left(\frac{r-x}{r} u_c + \frac{x}{r} u_d \right) \\ &= \frac{u_a - u_b - u_c + u_d}{rs} xy + \frac{-u_a + u_b}{r} x + \frac{-u_a + u_c}{s} y + u_a \quad \dots (4) \\ v_g(x, y) &= \frac{v_a - v_b - v_c + v_d}{rs} xy + \frac{-v_a + v_b}{r} x + \frac{-v_a + v_c}{s} y + v_a \end{aligned}$$

25

と書き換えることができる。従って、 $b_0 \sim b_7$ を伝送する代わりに $u_a, v_a, u_b, v_b, u_c, v_c, u_d, v_d$ を伝送しても同様の機能が実現できる。この様子を第1図に示す。現フレームの原画像102と参照画像101の間でグローバル動き補償が行われたとして、動きパラメータの代わりに代表点103、104、105、106の動きベクトル107、108、109、110（このとき、動きベクトルは現フレームの原画像の点を出発点として、参照画像内の対応する点を終点とするものとして定義する）を伝送しても良い。本明細書では式(1)を用いる方式を線形内・外挿に基づくグローバル動き補償、式(2)を用いる方式を共1次内・外挿に基づくグローバル動き補償とよぶ。

このグローバル動き補償の処理を、画像内のより小さい領域に適用するのがワーピング予測である。第2図に共一次内・外挿を用いるワーピング予測の例を示す。この図は、参照画像201を用いて現フレームの原画像202の予測画像を合成する処理を示したものである。まず現フレームは複数の多角形のパッチに分割され、パッチ分割された画像209となる。パッチの頂点は格子点とよばれ、各格子点は複数のパッチに共有される。例えば、パッチ210は、格子点211、212、213、214から構成され、これらの格子点は他のパッチの頂点を兼ねている。このように画像を複数のパッチに分割した後に、動き推定が行なわれる。ここに示す例では、動き推定は各格子点を対象として参照画像との間で行なわれる。この結果、動き推定後の参照画像203で各パッチは変形されたものとなる。例えば、パッチ210は、変形されたパッチ204に対応している。これは、動き推定の結果、格子点205、206、207、208がそれぞれ211、212、213、214に対応していると推定されたためである。このようにして格子点の動きベクトルを求め、共1次内挿によってパッチ内の各画素の動きベクトルを計算するこ

とにより、フレーム間予測画像が合成される。このワーピング予測の処理は基本的に第 1 図に示したグローバル動き補償と同じ処理であり、「画像の隅の点の動きベクトル」が「格子点の動きベクトル」に変えられているだけである。また、長方形の代わりに 3 角形のパッチを使用すれば、

5 線形内・外挿によるワーピング予測も実現することができる。

なお、画像全体の動きベクトル場を少ないパラメータを用いて表現するグローバル動き補償の処理を簡易にした符号化及び復号方法に関して本願出願人による発明「画像符号化方法および復号化方法」（特願平 8 - 6 0 5 7 2 号）及び「フレーム間予測画像の合成方法」（特願平 8 - 2 4 9 10 6 0 1 号）がある。

上述のグローバル動き補償やワーピング予測を導入することにより、画像の動きを少ないパラメータを用いて正確に表現することが可能となり、より高い情報圧縮率が実現できる。しかし、その一方で符号化及び復号化における処理量は従来の方式と比較して増加する。特に式(3)及び式(4)の除算は、処理を複雑にする大きな要因となってしまう。すなわち、グローバル動き補償やワーピング予測では、予測画像の合成のための処理量が多くなる問題が発生する。

発明の開示

20 発明の目的は、これらの動き補償方式における除算の処理をビット数の少ないレジスタを用いた 2 進数のシフト演算に置き換えることにより、演算量を減少させることにある。

上記目的を達成するため、本発明は、グローバル動き補償やワーピング予測によってフレーム間予測画像の合成処理を行う画像符号化及び復号化において、空間的な間隔が特徴を持つ複数の代表点の動きベクトル 25 に対し、2 段階の内・外挿入処理を行うことにより動きベクトルを求め

る。さらに詳しく言えば、フレーム間予測画像の合成処理において、

- 画素のサンプリング間隔を水平、垂直方向共に 1 として、サンプリング点が座標の水平、垂直成分が、共に整数に w を加えた数である点の上に存在している画像を対象として（ただし、 $w = w_n / w_d$ 、かつ w_n
- 5 は負ではない整数、かつ w_d は 2 の h_w 乗、かつ h_w は負ではない整数、かつ $w_n < w_d$ ）、4 個の代表点における動きベクトルに対し、共 1 次内・外挿を行うことによって画素の動きベクトルを計算する場合に、座標 (i, j) 、 $(i + p, j)$ 、 $(i, j + q)$ 、 $(i + p, j + q)$ に $(i, j, p, q$ は整数) に代表点が存在し、かつ代表点の動きベクトルの水平・垂直成分が $1 / k$ の整数倍の値をとり（ただし、 k は 2 の h_k 乗、かつ h_k は負ではない整数）、かつ座標 $(x + w, y + w)$ に位置する画素の動きベクトルを求めるときに、座標 (i, j) と $(i, j + q)$ [又は $(i + p, j)$] に位置する代表点の動きベクトルに対して線形内・外挿を行うことにより、座標 $(i, y + w)$ [又は $(x + w, j)$]
- 15 $(i, y + w)$ [又は $(x + w, j)$] に位置する点の動きベクトルの水平・垂直成分をそれぞれ $1 / z$ の整数倍をとる数値として（ただし、 z は 2 の h_z 乗、かつ h_z は負ではない整数）求め、さらに座標 $(i + p, j)$ [又は $(i, j + q)$] と $(i + p, j + q)$ に位置する代表点の動きベクトルに対して線形内・外挿を行うことにより、座標 $(i + p, y + w)$ [又は $(x + w, j + q)$]
- 20 $(i + p, y + w)$ [又は $(x + w, j + p)$] に位置する上記 2 個の動きベクトルに対して線形内・外挿を行うことにより、座標 $(x + w, y + w)$ に位置する画素の動きベクトルの水平・垂直成分をそれぞれ $1 / m$ の整数倍をとる数値として（ただし、 m は 2 の h_m 乗、かつ h_m は負ではない整数）求める。
- 25

本発明は、代表点の座標を巧みに選択することによって除算処理をシフト演算で実現できるようにし、さらにシフト演算においてシフトされるビット数を少なくすることにより、ビット数の少ないレジスタによって上記動き補償方式の演算が実現できる。

5

図面の簡単な説明

第1図は、代表点の動きベクトルを伝送するグローバル動き補償の例を示した図である。

第2図は、ワーピング予測の処理例を示した図である。

10 第3図は、高速な処理を行うための代表点の配置の例を示した図である。

第4図は、本発明のソフトウェア画像符号化装置の構成例を示した図である。

第5図は、本発明のソフトウェア画像復号化装置の構成例を示した図である。

15 第6図は、本発明による画像符号化装置の構成例を示した図である。

第7図は、本発明による画像復号化装置の構成例を示した図である。

第8図は、第6図の動き補償処理部616の構成例を示した図である。

第9図は、第6図の動き補償処理部616の他の構成例を示した図である。

第10図は、第7図の予測画像合成部711の構成例を示した図である。

20 第11図は、第9図の予測画像合成部1103の構成例を示した図である。

第 1 2 図は、グローバル動き補償予測画像合成部の構成例を示した図である。

第 1 3 図は、ソフトウェア画像符号化装置における処理のフローチャートの例を示した図である。

5 第 1 4 図は、ソフトウェア画像符号化装置における動き補償処理のフローチャートの例を示した図である。

第 1 5 図は、ソフトウェア画像復号化装置における処理のフローチャートの例を示した図である。

10 第 1 6 図は、ソフトウェア画像復号化装置における予測画像合成処理のフローチャートの例を示した図である

第 1 7 図は、2 段階の処理によりグローバル動き補償予測画像を合成する画像符号化・復号化を使用する装置の具体例を示した図である。

発明を実施するための最良の形態

15 本発明の理解を容易にするため、グローバル動き補償及びワーピング予測における演算の高速化方法に関する本出願人が先に出願した発明（特願平 08-060572 及び特願平 08-249601）を説明する。また、以下では本発明をグローバル動き補償に適用した場合に関して説明するが、本発明はグローバル動き補償と同様の処理を行うワーピング予測にも応用
20 することが可能である。

以下の説明では、画素のサンプリング間隔が水平・垂直方向共に 1 であるとして、座標の水平・垂直成分が共に整数に w を加えた値である点（ただし、 $w = w_n / w_d$ 、かつ w_n は負ではない整数、かつ w_d は正の整数、かつ $w_n < w_d$ ）に画素が存在しているとする。 w はグローバル
25 動き補償における代表点の座標と画素の座標の位相のずれを表しており、典型的な値としては 0、1/2、1/4 などが挙げられる。また、画像の

水平方向と垂直方向の画素数はそれぞれ r と s であり（ただし、 r と s は正の整数）、かつ画像の画素は水平座標が 0 以上 r 未満、垂直座標が 0 以上 s 未満の範囲に存在しているとする。

- 線形内・外挿（アフィン変換）又は共 1 次内・外挿（共 1 次変換）を用いた動き補償を行う際には、画素ごとの動きベクトルに対して量子化を行うと、ミスマッチの防止や演算の簡略化などの効果を得ることができる（特願平 06-193970）。以下では、画素の動きベクトルの水平成分と垂直成分が $1/m$ （ m は正の整数）の整数倍であるとする。また、「従来の技術」で説明した代表点の動きベクトルを用いるグローバル動き補償を行うと仮定し、各代表点の動きベクトルは $1/k$ （ k は正の整数）の整数倍であるとする。なお、本明細書では、「画素の動きベクトル」はグローバル動き補償を行う際に、実際に予測画像を合成するために用いる動きベクトルのことを指す。一方、「代表点の動きベクトル」は画素の動きベクトルを計算するために用いるパラメータを意味している。
- 従って、量子化ステップサイズの違いなどが原因で、同じ座標上に存在していても画素の動きベクトルと代表点の動きベクトルが一致しない場合も起こり得る。

- まず、共 1 次内・外挿を用いるグローバル動き補償について第 3 図を用いて説明する。この例では、第 1 図に示したように、代表点を画像 301 の隅に位置する点とはせず、 (i, j) 、 $(i+p, j)$ 、 $(i, j+q)$ 、 $(i+p, j+q)$ に位置する点 302、303、304、305 として一般化している（ i, j, p, q は整数）。このとき、点 302、303、304、305 は画像の内部に存在していても外部に存在していても良い。代表点の動きベクトルの水平・垂直成分を k 倍したものをそれぞれ (u_0, v_0) 、 (u_1, v_1) 、 (u_2, v_2) 、 (u_3, v_3) とすると $(u_0, v_0, u_1, v_1, u_2, v_2, u_3, v_3)$ は

整数)、 $(x+w, y+w)$ に位置する画素の動きベクトルの水平・垂直成分を m 倍したもの ($u(x+w, y+w), v(x+w, y+w)$) は、 $w=0$ のときは以下の式で表すことができる (ただし、 $x, y, u(x, y), v(x, y)$ は整数)。

5

$$\begin{aligned} u(x+w, y+w) &= u(x, y) \\ &= (((j+q-y)((i+p-x)u_0 + (x-i)u_1) \\ &\quad + (y-j)((i+p-x)u_2 + (x-i)u_3))m) // (pqk) \quad \dots (5) \\ v(x+w, y+w) &= v(x, y) \\ &= (((j+q-y)((i+p-x)v_0 + (x-i)v_1) \\ &\quad + (y-j)((i+p-x)v_2 + (x-i)v_3))m) // (pqk) \end{aligned}$$

10

ただし、「//」は通常の除算による演算結果が整数ではない場合にこれを近隣の整数に丸め込む除算で、演算子としての優先順位は乗除算と同等である。演算誤差を小さくするためには、非整数値は最も近い整数に丸め込まれることが望ましい。このとき整数に $1/2$ を加えた値の丸め

15 込み方法は、

- (1) 0 に近づける方向に丸め込む、
- (2) 0 から遠ざける方向に丸め込む、
- (3) 被除数が負の場合は 0 に近づける方向、正の場合は 0 から遠ざける方向に丸め込む (除数は常に正であるとする)、

20 (4) 被除数が負の場合は 0 から遠ざける方向、正の場合は 0 に近づける方向に丸め込む (除数は常に正であるとする)、

などが考えられる。これらの中で(3)と(4)は、被除数の正負に関わらず丸め込みの方向が変化しないため、正負判定が必要ない分だけ処理量の点で有利である。(3)を用いた高速処理は、例えば以下の式によって実現す

25 ることができる。

$$\begin{aligned}
 u(x+w, y+w) &= u(x, y) \\
 &= (Lpqk + ((j+q-y)((i+p-x)u_0 + (x-i)u_1) \\
 &\quad + (y-j)((i+p-x)u_2 + (x-i)u_3))m + ((pqk)\#2)) \\
 &\quad \#(pqk) - L
 \end{aligned}
 \quad \dots (6)$$

5

$$\begin{aligned}
 v(x+w, y+w) &= v(x, y) \\
 &= (Mpqk + ((j+q-y)((i+p-x)v_0 + (x-i)v_1) \\
 &\quad + (y-j)((i+p-x)v_2 + (x-i)v_3))m + ((pqk)\#2)) \\
 &\quad \#(pqk) - M
 \end{aligned}$$

ただし、「#」は小数点以下を0の方向に切り捨てる整数の除算であり、演算の優先順位は乗除算と同じであるとする。これは、一般に計算機では最も実現しやすい形式の除算である。ここで、LとMは除算の被除数を常に正に保つための数で、十分に大きな正の整数である。また、 $(pqk\#2)$ の項は、除算結果を最も近い整数に丸め込むために用いられる。

処理を整数化することはそれ自体処理量の低減に貢献するが、ここでp、q、kをそれぞれ2の α 、 β 、 h_k 乗（ α 、 β 、 h_k は負ではない整数）とすると、数5の除算は $\alpha + \beta + h_k$ ビットのシフト演算で実現できるため、計算機や専用ハードウェアにおける処理量を大きく減らすことができる。さらにmを2の h_m 乗とすれば（ h_m は負ではない整数、 $h_m < \alpha + \beta + h_k$ ）、式（6）は、

20

$$\begin{aligned}
 u(x+w, y+w) &= u(x, y) \\
 &= ((2L+1) << (\alpha + \beta + h_k - h_m - 1) \\
 &\quad + (j+q-y)((i+p-x)u_0 + (x-i)u_1) \\
 &\quad + (y-j)((i+p-x)u_2 + (x-i)u_3)) \\
 &\quad >> (\alpha + \beta + h_k - h_m) - L
 \end{aligned}
 \quad \dots (7)$$

$$v(x+w, y+w) = v(x, y)$$

25

$$\begin{aligned}
&= ((2M+1) \ll (\alpha + \beta + h_k - h_m - 1) \\
&\quad + (j+q-y)((i+p-x)v_0 + (x-i)v_1) \\
&\quad + (y-j)((i+p-x)v_2 + (x-i)v_3)) \\
&\gg (\alpha + \beta + h_k - h_m) - M
\end{aligned}$$

- と書き換えることができ（「 $x \ll \alpha$ 」は x を α ビット左にシフトして下
 5 位 α ビットに0を入れる、「 $x \gg \alpha$ 」は x を α ビット右にシフトして上
 位 α ビットに0を入れることを意味し、これらの演算子の優先順位は加
 減算と乗除算の間であるとする）、シフトされるビット数を $\alpha + \beta +$
 $h_k - h_m$ とすることができる。

- w が0ではないときには、 $w = w_n / w_d$ の定義にしたがい、式（5）
 10 は以下のように書き換えることができる。

$$\begin{aligned}
u(x+w, y+w) &= u\left(x + \frac{w_d}{w_n}, y + \frac{w_d}{w_n}\right) \\
&= (((w_d j + w_d q - w_d y - w_n) ((w_d i + w_d p - w_d x - w_n) u_0 \\
&\quad + (w_d x + w_n - w_d i) u_1) \\
&\quad + (w_d y + w_n - w_d j) ((w_d i + w_d p - w_d x - w_n) u_2 \\
&\quad + (w_d x + w_n - w_d i) u_3)) m) \\
&\quad // (w_d^2 p q k) \quad \dots (8) \\
15 \quad v(x+w, y+w) &= v\left(x + \frac{w_d}{w_n}, y + \frac{w_d}{w_n}\right) \\
&= (((w_d j + w_d q - w_d y - w_n) ((w_d i + w_d p - w_d x - w_n) v_0 \\
&\quad + (w_d x + w_n - w_d i) v_1) \\
&\quad + (w_d y + w_n - w_d j) ((w_d i + w_d p - w_d x - w_n) v_2 \\
&\quad + (w_d x + w_n - w_d i) v_3)) m) \\
&\quad // (w_d^2 p q k)
\end{aligned}$$

20

- このとき、 w_d が2の h_w 乗であり、かつ h_w は負ではない整数であ
 るとすれば、 $(p \cdot q \cdot k \cdot w_d \cdot w_d)$ による除算は $\alpha + \beta + h_k + 2$
 h_w ビットのシフト演算となり、 $w = 0$ の場合と同様に除算をシフト演
 算に置換することが可能となる。また、式(7)の場合と同様に、 h_m
 25 $< \alpha + \beta + h_k + 2 h_w$ であれば、分母、分子の両方を m で割ること
 によってシフトされるビット数を $\alpha + \beta + h_k + 2 h_w - h_m$ ビットに減

らすことが可能となる。このように、 $w d$ が 2 の $h w$ 乗であれば、 $w = 0$ の場合の処理と $w \neq 0$ の場合の処理は本質的に同じである。以下本明細書では、数式が多少複雑となるが、 $w \neq 0$ の場合について検討を行う。 $w = 0$ の場合の計算結果を求めるためには、 $w n = 0$ 、 $w d = 1$ 、 $h w = 0$ を代入すれば良い。

送信側と受信側で同じグローバル動き補償予測画像を得るためには、代表点の動きベクトルに関する情報を何らかの形で受信側に伝える必要がある。代表点の動きベクトルそのまま伝送する方法もあるが、画像の隅の点の動きベクトルを伝送し、この値から代表点の動きベクトルを計算する方法もある。この方法に関し、以下に説明する。

画像の隅の 4 個の点 $(-c, -c)$ 、 $(r-c, -c)$ 、 $(-c, s-c)$ 、 $(r-c, s-c)$ の動きベクトルが $1/n$ 整数倍の値のみとれるとして (n は正の整数、 $c = c_n / c_d$ 、かつ c_n は負ではない整数、かつ c_d は正の整数、かつ $c_n < c_d$)、これらの水平・垂直成分を n 倍した (u_{00}, v_{00}) 、 (u_{01}, v_{01}) 、 (u_{02}, v_{02}) 、 (u_{03}, v_{03}) がグローバル動きパラメータとして伝送されたとする。 c は画像の隅の点と代表点の間の位相のずれを表している。この c の典型的な値としては 0 、 $1/2$ 、 $1/4$ などが挙げられる。このとき、点 (i, j) 、 $(i+p, j)$ 、 $(i, j+q)$ 、 $(i+p, j+q)$ それぞれの動きベクトルの水平・垂直成分を k 倍したものである (u_0, v_0) 、 (u_1, v_1) 、 (u_2, v_2) 、 (u_3, v_3) を、

$$\begin{aligned} u_0 &= u'(i, j) \\ v_0 &= v'(i, j) \\ u_1 &= u'(i+p, j) \\ v_1 &= v'(i+p, j) \\ u_2 &= u'(i, j+q) \\ v_2 &= v'(i, j+q) \\ u_3 &= u'(i+p, j+q) \\ v_3 &= v'(i+p, j+q) \end{aligned} \quad \dots (9)$$

と定義する。ただし、 $u'(x, y)$ 、 $v'(x, y)$ は、式(5)を変

5 形して、

$$\begin{aligned} u'(x, y) &= (((c_d s - c_n - c_d y) ((c_d r - c_n - c_d x) u_{00} + (c_d x + c_n) u_{01}) \\ &\quad + (c_d y + c_n) ((c_d r - c_n - c_d x) u_{02} + (c_d x + c_n) u_{03})) k) \\ &\quad /// (c_d^2 r s n) \quad \dots (10) \\ v'(x, y) &= (((c_d s - c_n - c_d y) ((c_d r - c_n - c_d x) v_{00} + (c_d x + c_n) v_{01}) \\ &\quad + (c_d y + c_n) ((c_d r - c_n - c_d x) v_{02} + (c_d x + c_n) v_{03})) k) \\ &\quad /// (c_d^2 r s n) \end{aligned}$$

10

と定義する。このとき、「///」は通常の除算による演算結果が整数ではない場合にこれを近隣の整数に丸め込む除算で、演算子としての優先順位は乗除算と同等である。こうして (u_0, v_0) 、 (u_1, v_1) 、 $(u$

15 $2, v_2)$ 、 (u_3, v_3) を計算し、 (i, j) 、 $(i + p, j)$ 、 $(i, j + q)$ 、 $(i + p, j + q)$ を代表点とするグローバル動き補償を行えば、 $(-c, -c)$ 、 $(r - c, -c)$ 、 $(-c, s - c)$ 、 $(r - c, s - c)$ を代表点とするグローバル動き補償を近似することができる。

このときに、上で述べたように p と q を 2 の負ではない整数乗とすれば、処理を簡略化することが可能となる。一般的に、式(5)に示したような計算によって画像内の画素の動きベクトルを求めるときには、外挿の処理を行わないようにすることが望ましい。これは、外挿処理によって代表点の動きベクトルの量子化誤差を増幅しないようにするためである。以上の理由から、代表点は画像内の画素をすべて囲むような形に

25 配置することが望ましい。従って、 $i = j = c = 0$ の場合などは p と q は r と s とほぼ同じか、やや大きめの値をとするのが適当である。しか

し、 p と q の値をあまり大きくし過ぎると演算に必要なビット数が増加してしまうので注意が必要である。

式(9)、式(10)の処理において演算誤差を小さくするためには、

「 $///$ 」は非整数値を最も近い整数に丸め込むことが望ましい。このとき

- 5 整数に $1/2$ を加えた値の丸め込み方法としては、上で述べた(1)~(4)の方法が考えられる。ただし、式(5)（画素ごとに計算）の場合と比較して、式(14)（1枚の画像で4回のみ計算）は演算が実行される回数が少ないため、式(1)又は(2)の方法を選んだとしても全体の演算量に大きな影響は与えない。

- 10 上述の例のように、 p と q の値が2の負ではない整数乗となるようにすれば、グローバル動き補償におけるフレーム間予測画像の合成処理は大幅に簡略化することができる。しかし、ここでもう1つの問題が発生する。例えば、画像符号化における典型的なパラメータとして $p = 512$ 、 $q = 512$ 、 $k = 32$ 、 $m = 16$ 、 $w_d = 2$ 、 $w_n = 1$ ($w = 0.5$)
- 15 である場合を考えると、 $\alpha + \beta + h_k + 2h_w - h_m = 21$ となる。このことは、 $u(x+w, y+w)$ が2進数で12ビット以上を必要とする値である場合には、式(8)の演算を高速に実行するために33ビット以上のレジスタが必要になることを意味している。 $m = 16$ である場合などには、 $u(x+w, y+w)$ は実際の動きベクトルの水平成分に16
- 20 を掛けた値となるため、これが2進数で12ビット以上必要な値となるケースは十分にあり得る。しかし、その一方で33ビット以上の整数を格納できるレジスタを持つプロセッサは現時点では少なく、一つ将来的にも高価となることが予想される。また、一般的にプロセッサの回路規模が大きくなれば、その分だけ消費電力も多くなるため、大きなレジス
- 25 タを要求するアルゴリズムは消費電力の観点からも不利となる。従って、除算をシフト演算に置換できた場合でも、シフトされるビット数はでき

るだけ少ないことが望ましい。

- この問題を解決するため本発明では、以下に説明する２段階の処理によるアルゴリズムをとる。 (i, j) 、 $(i + p, j)$ 、 $(i, j + q)$ 、 $(i + p, j + q)$ に位置する代表点の動きベクトルを用いて $(x + w, y + w)$ に位置する画素の動きベクトルを計算する前に、まず $(i, y + w)$ と $(i + p, y + w)$ に存在する仮代表点の動きベクトルを、水平・垂直成分が $1/z$ の整数倍（ z は正の整数）となるように求める。上の例と同様に代表点 (i, j) 、 $(i + p, j)$ 、 $(i, j + q)$ 、 $(i + p, j + q)$ の動きベクトルの水平・垂直成分を k 倍したものをそれぞれ (u_0, v_0) 、 (u_1, v_1) 、 (u_2, v_2) 、 (u_3, v_3) とする（ $u_0, v_0, u_1, v_1, u_2, v_2, u_3, v_3$ は整数）。このとき、 $(i, y + w)$ と $(i + p, y + w)$ に仮代表点を配置し、これらの仮代表点の動きベクトルの水平・垂直成分を z 倍したものである $(u_L(y + w), v_L(y + w))$ と $(u_R(y + w), v_R(y + w))$ を、以下のように定義する。

$$\begin{aligned}
 u_L(y+w) &= (((w_dj + w_dq - w_dy - w_n) u_0 + (w_dy + w_n - w_dj) u_2) z) \text{////} (w_dqk) \\
 v_L(y+w) &= (((w_dj + w_dq - w_dy - w_n) v_0 + (w_dy + w_n - w_dj) v_2) z) \text{////} (w_dqk) \\
 u_R(y+w) &= (((w_dj + w_dq - w_dy - w_n) u_1 + (w_dy + w_n - w_dj) u_3) z) \text{////} (w_dqk) \quad \dots (11) \\
 v_R(y+w) &= (((w_dj + w_dq - w_dy - w_n) v_1 + (w_dy + w_n - w_dj) v_3) z) \text{////} (w_dqk)
 \end{aligned}$$

- このとき、「////」は通常を除算による演算結果が整数ではない場合にこれを近隣の整数に丸め込む除算で、演算子としての優先順位は乗除算と同等である（この「////」には、上で説明した「///」と同様の機能が要求

- される)。(i, y+w)は(i, j)と(i, j+q)を結んだ線上に存在しているため、(u_L(y+w), v_L(y+w))は、(u₀, v₀)と(u₂, v₂)を用いた1次元の線形内・外挿で容易に求めることができる。また、同様に(i+p, y+w)は(i+p, j)と(i+p, j+q)を結んだ線上に存在しているため、同じように1次元の線形内・外挿で求めることができる。

- このようにして求めた仮代表点の動きベクトル(u_L(y+w), v_L(y+w))と(u_R(y+w), v_R(y+w))に対して1次元の線形内・外挿を行うことにより、(x+w, y+w)に存在する画素の動きベクトルの水平・垂直成分をm倍したものである(u(x+w, y+w), v(x+w, y+w))を求める。この処理は、以下の式(12)に従って行われる。

$$\begin{aligned} u(x+w, y+w) &= (((w_d i + w_d p - w_d x - w_n) u_L(y+w) \\ &\quad + (w_d x + w_n - w_d i) u_R(y+w)) m) // (w_d p z) \\ v(x+w, y+w) &= (((w_d i + w_d p - w_d x - w_n) v_L(y+w) \\ &\quad + (w_d x + w_n - w_d i) v_R(y+w)) m) // (w_d p z) \end{aligned} \quad \cdots (12)$$

- ここでも上と同様にpを2の α 乗、mを2の h_m 乗、zを2の h_z 乗、 w_d を2の h_w 乗(α 、 h_m 、 h_z 、 w_d は負ではない整数)とすれば、式(12)における $p \cdot z \cdot w_d$ による除算は、 $\alpha + h_z + h_w - h_m$ ビットの右シフト(ただし、 $h_m < \alpha + h_z + h_w$ の場合)に置換することができる。しかも、 $z = 16$ ($h_z = 4$)とした上で、上で述べた典型的なパラメータ $p = 512$ 、 $q = 512$ 、 $k = 32$ 、 $m = 16$ 、 $w_d = 2$ 、 $w_n = 1$ ($w = 0.5$)を使用した場合、シフトされるビット数は10ビットとなり、演算に用いるレジスタに必要なビット数を大幅に抑えることが可能となる。なお、上の例では、まず代表点の動きベクトルに対して垂直方向の1次元線形内・外挿を行って仮代表点の動きベクトルを

求め、この仮代表点の動きベクトルに対して水平方向の1次元線形内・外挿を行って画素の動きベクトルを求めている。これとは逆に、仮代表点の動きベクトルを求める際には水平方向、画素の動きベクトルを求める際には垂直方向の1次元線形内・外挿を行っても同様の機能を実現することができる。

この方式では、画素の動きベクトルを求める際に式(11)と式(12)の2段階の処理が必要となるため、一見演算量が多くなるように思われる。しかし、一旦仮代表点の動きベクトルを求めてしまえば、これが垂直座標 $y + w$ に存在しているライン上の r 個の画素すべてに対して使用できるため、全体の処理量の中に占める式(11)の処理量はきわめて少なくなる。従って、シフトされるビット数の削減によって得られる利益(=より小さいレジスタの活用)の影響の方が、式(11)の計算を実行する分の演算量の増加による悪影響より大きくなる。

上記処理により $(u(x+w, y+w), v(x+w, y+w))$ の値が得られた後には、以下の処理によって $(u(x+w, y+w), v(x+w, y+w))$ を整数部 $(u_I(x+w, y+w), v_I(x+w, y+w))$ と小数部 $(u_F(x+w, y+w), v_F(x+w, y+w))$ に分けることができる。

$$\begin{aligned} u_I(x+w, y+w) &= ((Lm + u(x+w, y+w)) \gg h_m) - L \\ v_I(x+w, y+w) &= ((Mm + v(x+w, y+w)) \gg h_m) - M \end{aligned} \quad \dots (13)$$

$$\begin{aligned} u_F(x+w, y+w) &= u(x+w, y+w) - u_I(x+w, y+w) m \\ v_F(x+w, y+w) &= v(x+w, y+w) - v_I(x+w, y+w) m \end{aligned} \quad \dots (14)$$

ただし、 $u_I(x+w, y+w)$ と $v_I(x+w, y+w)$ は整数であり、画素の動きベクトルの整数部を表している。一方、 $u_F(x+w, y+w)$ と $v_F(x+w, y+w)$ はそれぞれ0以上 m 未満の値を持つ整数であり、画素の動きベクトルの小数部を m 倍したものである。なお、上の

例と同様に m は2の h_m 乗であり(h_m は負ではない整数)、 L と M はシフトされる値を負の値ではなくするための十分に大きな整数である。

輝度値の内挿方式として共1次内挿が用いられる場合には、さらに以下の処理によってフレーム間予測画像内の画素の輝度値が求められる。

- 5 $x' = x + w + u I(x + w, y + w)$ 、 $y' = y + w + v I(x + w, y + w)$ として、参照画像の (x', y') 、 $(x' + 1, y')$ 、 $(x', y' + 1)$ 、 $(x' + 1, y' + 1)$ に位置する画素の輝度値をそれぞれ Y_a 、 Y_b 、 Y_c 、 Y_d とすれば、フレーム間予測画像において $(x + w, y + w)$ に位置する画素の輝度値 $Y(x + w, y + w)$ は

$$10 \quad Y(x+w, y+w) = ((m-v_F)((m-u_F)Y_a + u_F Y_b) + v_F((m-u_F)Y_c + u_F Y_d) + (m^2 >> 1)) >> (2h_m) \dots (15)$$

によって求められる。ただし、 u_F 、 v_F はそれぞれ $u_F(x + w, y + w)$ 、 $v_F(x + w, y + w)$ の略号である。

- 式(12)と式(13)では、それぞれ $\alpha + h_z + h_w - h_m$ ビットと h_m ビットの右シフトが行われる。このことは、式(10)の計算の際に $(\alpha + h_z + h_w - h_m) + h_m = \alpha + h_z + h_w$ ビットのシフトを行えば一気に $u I(x + w, y + w)$ と $v I(x + w, y + w)$ を求めることができることを意味する。このとき、 $\alpha + h_z + h_w$ を8の整数倍とすると、実装上便利である。一般的にプロセッサのレジスタは8ビット単位
- 15 の大きさを持っており、8ビットのレジスタを2個(上位ビットのレジスタと下位ビットのレジスタ)つなげて16ビットのレジスタとして使用したり、8ビットのレジスタを4個、又は16ビットのレジスタを2個つなげて32ビットのレジスタとして使用することができるようになっている場合が多い。ここで例えば16ビットのシフト演算によって
- 20 $u I(x + w, y + w)$ と $v I(x + w, y + w)$ の値が計算されるのであれば、わざわざシフト演算を行う必要はなくなる。つまり、シフトされ

る前の値を32ビットのレジスタに格納しておき、その上位16ビットを独立したレジスタとして使用すれば、その16ビットレジスタに $uI(x+w, y+w)$ 又は $vI(x+w, y+w)$ の値が格納されていることになる。

- 5 もちろん、シフトされるビット数を8の整数倍とすることは、式(10)の処理だけでなく本明細書でこれまで述べてきたあらゆるシフト演算に対し、実装を容易にする効果を持つ。しかし、特に実行される回数の多いシフト演算（例えば画素ごとに実行されるシフト演算）に対して実装を容易にすることは重要である。また、シフトされるビット数が8の整数倍ではない場合でも、分母と分子に事前に同じビット数だけの左シフトを加えておくことにより、除算による右シフトを増やすことは可能である。例えば、6ビットの右シフトによって実現される演算があった場合に、シフトされる数値にあらかじめ4を掛けておく（これは2ビットの左シフトを行ったことに相当する）ことにより、同じ演算を8ビット
- 10 の右シフトとして実現することが可能となる式(5)の $u(x+w, y+w)$ に関する式を例にとれば、あらかじめ u_0, u_1, u_2, u_3 を4倍しておくことにより、この処理を実現することが可能となる）。ただし、このような処理を行う際には、シフトされる数に関してオーバーフローが発生しないように注意する必要がある。
- 15 画像符号化装置及び画像復号化装置には、複数の画像サイズに対応できるようにになっているものが多い。この場合、例えば式(12、13、14)を用いたグローバル動き補償を実行したときには、画像サイズの変化に応じてシフトされるビット数が変化する現象が起こり、シフトされるビット数を8の整数倍に固定しておくことができなくなる。このような
- 20 場合、次に述べるような対処法がある。例えば、上の例のように $uI(x+w, y+w)$ と $vI(x+w, y+w)$ を求めるために $\alpha + h_z + h_w$

- ビットの右シフトが必要であり、 α が7～11の値をとり得る場合を考える。このとき、 α が10より小さいときは $h_z=5$ 、 $h_w=1$ 、 $\alpha=11$ のときは $h_z=4$ 、 $h_w=1$ とすれば、シフトされるビット数を常に16以下とすることができる。上で述べたように、シフトされるビット数が16より小さい場合には、あらかじめシフトされる数に定数を掛けておくことにより、シフトされるビット数を擬似的に16ビットとすることが可能である。このように、画像サイズが変化したときに他のパラメータ（例えば動きベクトルの量子化ステップサイズ）もこれに合わせて変化させることにより、シフトされるビット数が都合の良い値となるように制御することができる。しかし、上記の方法を使う場合には、復号画像の画質に著しい劣化を生じさせるほど、動きベクトルの量子化ステップサイズを大きくしてしまわないように注意する必要がある。

- 一般的なグローバル動き補償に本明細書で示したアルゴリズムを適用した場合には、まず $1/n$ 画素精度の画像の隅の点の動きベクトルを用いて代表点の動きベクトルを $1/k$ 画素精度で求め、続いて代表点の動きベクトルを用いて仮代表点の動きベクトルを $1/z$ 画素精度で求めた後に、この仮代表点の動きベクトルを用いて画素の動きベクトルが $1/m$ 画素精度で求められる。画像の隅の点の動きベクトルが動きパラメータとして伝送される場合には、このパラメータによる共1次内・外挿を正確に近似するという意味で、 k をできるだけ大きな値にすることが望ましい。しかし、いずれにせよ代表点の動きベクトルの水平・垂直成分には、量子化の影響で $1/(2k)$ 以下の絶対値をもつ誤差が含まれることになる。近似を正確にするという意味からは、仮代表点の動きベクトルも、なるべく精度を高くすることが望ましい。しかし、仮代表点の動きベクトルは代表点の動きベクトルを用いて求められるため、代表点の動きベクトル以上の精度を持たせて計算してもあまり意味がない。従

って、演算に必要なビット数を抑える意味で $z \leq k$ とすることが望ましい。また、同様の理由により、 $m \leq z$ とすることが望ましい。

- 5 これまで共一次内・外挿を用いたグローバル動き補償に関して説明してきたが、線形内・外挿を用いた場合も同様の処理を導入することによって、シフトされるビット数を制御することができる。例えば、 (i, j) 、 $(i+p, j)$ 、 $(i, j+q)$ に存在する $(i, j, p, q$ は整数) 代表点の動きベクトルの水平・垂直成分を k 倍したものをそれぞれ (u_0, v_0) 、 (u_1, v_1) 、 (u_2, v_2) とする $(u_0, v_0, u_1, v_1, u_2, v_2$ は整数)。このとき、画素 $(x+w, y+w)$ の
- 10 動きベクトルの水平・垂直成分を m 倍したもの $(u(x+w, y+w), v(x+w, y+w))$ は以下の式 (16) で表すことができる (ただし、 $x, y, u(x+w, y+w), v(x+w, y+w)$ は整数、 w の定義は上と同じ)。

$$\begin{aligned}
 u(x+w, y+w) &= (((u_1-u_0)(w_d x+w_n-w_d i) q \\
 &\quad + (u_2-u_0)(w_d y+w_n-w_d j) p + u_0 w_d p q) m) \\
 &\quad // (w_d p q k) \quad \dots (16) \\
 v(x+w, y+w) &= (((v_1-v_0)(w_d x+w_n-w_d i) q \\
 &\quad + (v_2-v_0)(w_d y+w_n-w_d j) p + v_0 w_d p q) m) \\
 &\quad // (w_d p q k)
 \end{aligned}$$

- この場合も p, q, k, m, w_d がそれぞれ 2 の α 乗、 β 乗、 $h k$ 乗、 $h m$ 乗、 $h w$ 乗であり ($\alpha, \beta, h k, h m, h w$ は負ではない整数)、
- 20 さらに $\alpha \geq \beta$ であるとすれば、この式は

$$\begin{aligned}
 u(x+w, y+w) &= (((u_1-u_0)(w_d x+w_n-w_d i) 2^{\alpha-\beta} \\
 &\quad + (u_2-u_0)(w_d y+w_n-w_d j) + u_0 w_d p) m) // (w_d p k) \\
 v(x+w, y+w) &= (((v_1-v_0)(w_d x+w_n-w_d i) 2^{\alpha-\beta} \\
 &\quad + (v_2-v_0)(w_d y+w_n-w_d j) + v_0 w_d p) m) // (w_d p k) \quad \dots (17)
 \end{aligned}$$

- と書き換えることができ、共 1 次内・外挿を用いた場合と同様に、 $\alpha + h k + h w$ ビットの右シフトにより $(x + w, y + w)$ に存在する画素の動きベクトルの整数部を求めることができる。従って、 $\alpha + h k + h w$ が 8 の整数倍となるようにすれば、上と同様の理由により実装を行い
- 5 やすくすることができる。なお、 $\alpha < \beta$ の場合には、シフトされるビット数は $\beta + h k + h w$ ビットとなる。

以下、上記フレーム間予測画像の合成処理を採用する本発明の符号化方法及び復号化方法を実施する画像符号化装置、復号化装置の構成について述べる。

- 10 第 6 図は、本発明によるに画像符号化装置の一実施形態の構成を示す。同図において、動き補償処理部 6 1 6 を除いては、従来知られている画像符号化装置と実質的に同じである。

- 減算器 6 0 2 は入力画像（符号化しようとする現フレームの原画像）6 0 1 とフレーム間／フレーム内符号化切り換えスイッチ 6 1 9 の出力
- 15 画像 6 1 3（フレーム間予測画像）との差を計算し、誤差画像 6 0 3 を出力する。この誤差画像は、D C T 変換器 6 0 4 で D C T 係数に変換された後に量子化器 6 0 5 で量子化され、量子化 D C T 係数 6 0 6 となる。この量子化 D C T 係数は伝送情報として通信路に出力されると同時に、符号化器内でもフレーム間予測画像を合成するために使用される。

- 20 以下にフレーム予測画像合成の手順を説明する。量子化 D C T 係数 6 0 6 は、逆量子化器 6 0 8 と逆 D C T 変換器 6 0 9 を経て復号誤差画像 6 1 0（受信側で再生される誤差画像と同じ画像）となる。これに、加算器 6 1 1 においてフレーム間／フレーム内符号化切り換えスイッチ 6 1 9 の出力画像 6 1 3（後述）が加えられ、現フレームの復号画像 6 1
- 25 2（受信側で再生される現フレームの復号画像と同じ画像）を得る。この画像は一旦フレームメモリ 6 1 4 に蓄えられ、1 フレームの時間だけ

遅延される。従って、現時点では、フレームメモリ 6 1 4 は前フレームの復号画像 6 1 5 を出力している。この前フレームの復号画像と現フレームの入力画像 6 0 1 が動き補償処理部 6 1 6 に入力される。動き補償処理部 6 1 6 は前述のフレーム間予測画像の合成を行う。その構成については後で述べる。

予測画像 6 1 7 は、「0」信号 6 1 8 と共にフレーム間／フレーム内符号化切り換えスイッチ 6 1 9 に入力される。このスイッチは、両入力のいずれかを選択することにより、フレーム間符号化とフレーム内符号化を切り換える。予測画像 6 1 7 が選択された場合（第 6 図はこの場合を表している）には、フレーム間符号化が行われる。一方、「0」信号が選択された場合には、入力画像がそのまま D C T 符号化されて通信路に出力されるため、フレーム内符号化が行われる。受信側が正しく復号化画像を得るためには、送信側でフレーム間符号化が行われたかフレーム内符号化が行われたかを知る必要がある。このため、識別フラグ 6 2 1 が通信路へ出力される。最終的な H. 2 6 1 符号化ビットストリーム 6 2 3 は多重化器 6 2 2 で量子化 D C T 係数、動きベクトル、フレーム内／フレーム間識別フラグの情報を多重化することによって得られる。

第 7 図は、第 6 図の符号化器が出力した符号化ビットストリームを受信する復号化器 7 0 0 の構成例を示す。受信したビットストリーム 7 1 7 は、分離器 7 1 6 で量子化 D C T 係数 7 0 1、動きベクトル 7 0 2、フレーム内／フレーム間識別フラグ 7 0 3 に分離される。量子化 D C T 係数 7 0 1 は逆量子化器 7 0 4 と逆 D C T 変換器 7 0 5 を経て復号化された誤差画像 7 0 6 となる。この誤差画像は加算器 7 0 7 でフレーム間／フレーム内符号化切り換えスイッチ 7 1 4 の出力画像 7 1 5 を加算され、復号化画像 7 0 8 として出力される。フレーム間／フレーム内符号化切り換えスイッチはフレーム間／フレーム内符号化識別フラグ 7 0 3

に従って、出力を切り換える。フレーム間符号化を行う場合に用いる予測画像 7 1 2 は、予測画像合成部 7 1 1 において合成される。ここでは、フレームメモリ 7 0 9 に蓄えられている前フレームの復号画像 7 1 0 に対して、受信した動きベクトル 7 0 2 に従って位置を移動させる処理が行われる。一方フレーム内符号化の場合、フレーム間／フレーム内符号化切り換えスイッチは、「0」信号 7 1 3 をそのまま出力する。

第 8 図は、代表点の動きベクトルを伝送する線形内・外挿に基づくグローバル動き補償方式を採用した画像符号化器の動き補償処理部 6 1 6 の構成例を示す。第 6 図と同じ番号は同じものを指す。グローバル動き推定部 8 0 2 で前フレームの復号画像 6 1 5 と現フレームの原画像 6 0 1 との間でグローバル動き補償に関する動き推定が行われ、グローバル動き補償のパラメータ（例えば、上記 u_a 、 v_a 、 u_b 、 v_b 、 u_c 、 v_c 、 u_d 、 v_d の値）が推定される。これらの値に関する情報 8 0 3 は動き情報 6 2 0 の一部として伝送される。グローバル動き補償の予測画像 8 0 4 は式(3)を用いてグローバル動き補償予測画像合成部 8 0 8 で合成され、ブロックマッチング部 8 0 5 に供給される。ここでは、グローバル動き補償の予測画像と現フレームの原画像との間でブロックマッチングによる動き補償（動き推定と予測画像合成）が行われ、ブロックの動きベクトル情報 8 0 6 と最終的な予測画像 6 1 7 が得られる。この動きベクトル情報は動きパラメータ情報と多重化部 8 0 7 において多重化され、動き情報 6 2 0 として出力される。

第 10 図は、第 7 図の予測画像合成部 7 1 1 の構成例を示す。他の図と同じ番号は同じものを指す。前フレームの復号画像 7 1 0 に対し、動き情報 7 0 2 から分割部 1 0 0 2 において抽出されたグローバル動き補償パラメータ 8 0 3 を用いて、グローバル動き補償予測画像合成部 8 0 8 においてグローバル動き補償の予測画像 8 0 4 が合成される。画像 8

04はブロックマッチング予測画像合成部1001に供給され、動き情報702から抽出されたブロックマッチングの動きベクトル情報806を用いて最終的な予測画像712が合成される。

第9図は、動き補償処理部616の他の構成例をに示す。第6図と同じ番号は同じものを指す。この例では、各ブロックに関してグローバル動き補償かブロックマッチングのいずれかが適用される。前フレームの復号画像615と現フレームの原画像601との間で、グローバル動き推定部902、グローバル動き補償予測画像合成部911ではグローバル動き補償、ブロックマッチング部905ではブロックマッチングにより、それぞれ独立に動き補償の処理が行われる。選択スイッチ908は、グローバル動き補償による予測画像903とブロックマッチングによる予測画像906の間でブロックごとに最適な方式を選択する。グローバル動き補償パラメータ904、ブロックごとの動きベクトル907、グローバル動き補償／ブロックマッチングの選択情報909は多重化部910で多重化され、動き情報620として出力される。

第11図は、動き補償処理部901を用いる画像符号化器が生成するビットストリームを復号化する復号化器の、予測画像合成部1103の構成例を示す。他の図と同じ番号は同じものを指す。前フレームの復号画像710に対し、動き情報702から分割部1102において抽出されたグローバル動き補償パラメータ904を用いて、グローバル動き補償予測画像合成部911においてグローバル動き補償の予測画像903が合成される。また、これとは独立に前フレームの復号画像710に対し、動き情報702から抽出されたブロックマッチングの動きベクトル情報907を用いてブロックマッチング予測画像合成部1101においてブロックマッチングの予測画像906が合成される。選択スイッチ1104は、グローバル動き補償による予測画像903とブロックマッ

ングによる予測画像 906 の間で、動き情報 702 から抽出された選択情報 909 に基づいて、ブロックごとに一方の方式を選択する。このブロックごとの選択処理を経て、最終的な予測画像 712 が合成される。

- 第 12 図は、上述の本発明によるグローバル動き補償予測画像合成部の機能的構成を示す。グローバル動き補償パラメータとして、画像の隅の点の動きベクトルが伝送されたとする。この画像の隅の点の動きベクトルに関する情報 1204 を用いて演算処理部 1205 において式(9)、(10)を用いて代表点の動きベクトルが計算される。この代表点の動きベクトルに関する情報 1206 を用いて演算処理部 1207 では、式(11)を用いてライン（垂直座標が共通の値である画素）ごとに仮代表点の動きベクトルが計算される。さらにこの仮代表点の動きベクトルに関する情報 1208 を活用して演算処理部 1209 では画素ごとの動きベクトルが式(12)により計算される。一方、処理部 1211 では、この画素ごとの動きベクトルに関する情報 1210 と、前フレームの復号画像 1202 を用いてグローバル動き補償の予測画像 1203 が合成、出力される。

本発明は、専用回路・専用チップを用いる画像符号化装置、画像復号化装置の他に、汎用プロセッサを用いるソフトウェア画像符号化装置、ソフトウェア画像復号化装置にも適用することができる。

- 図 4 及び図 5 は、それぞれソフトウェア画像符号化装置 400 とソフトウェア画像復号化装置 500 の例を示す。ソフトウェア符号化装置 400 では、入力画像 401 は、入力フレームメモリ 402 に蓄えられ、汎用プロセッサ 403 は入力フレームメモリ 402 から情報を読み込んで符号化の処理を行う。汎用プロセッサ 403 を駆動するためのプログラムはハードディスクやフロッピーディスクなどによる蓄積デバイス 408 から読み出されてプログラム用メモリ 404 に蓄えられる。また、

汎用プロセッサ 403 は、処理用メモリ 405 を活用して符号化の処理を行う。汎用プロセッサ 403 が出力する符号化情報は、一旦出力バッファ 406 に蓄えられた後に符号化ビットストリーム 407 として出力される。

- 5 図 13 は、図 4 に示したソフトウェア符号化器上で動作する符号化ソフトウェアのフローチャートを示す。まずステップ 1301 で画像符号化処理が開始され、ステップ 1302 で変数 N に 0 が代入される。続いてステップ 1303、ステップ 1304 で N の値が 100 である場合には、0 が代入される。N はフレーム数のカウンタであり、1 枚のフレーム
- 10 の処理が終了する度に 1 が加算され、符号化を行う際には 0 ~ 99 の値をとることが許される。N の値が 0 であるときには符号化中のフレームは I フレーム（動き補償は行わず、全てのブロックでフレーム内符号化が行われるフレーム）であり、それ以外のときは P フレーム（動き補償を行うブロックの存在するフレーム）となる。N の値が 100 である
- 15 ことは、P フレーム が 99 枚符号化された後に I フレームが 1 枚符号化されることを意味している。N の最適な値は符号化器の性能や符号化器が使用される環境により変化する。この例では 100 という値を使用した。これは N の値が必ず 100 でなければならないことを意味しているわけではない。フレームタイプ（I 又は P）の決定と出力はステップ 1
- 20 305 で行われる。N の値が 0 である場合にはフレームタイプの識別情報として 'I' が出力バッファに出力され、これから符号化処理を行うフレームは I フレームとなる。なお、ここで「出力バッファに出力される」とは、出力バッファ（第 4 図の 406）に蓄えられた後に符号化ビットストリームの一部として符号化装置から外部に出力される。N が 0
- 25 ではない場合には、フレームタイプの識別情報として 'P' が出力バッファに出力され、これから符号化処理を行うフレームは P フレームとな

- る。ステップ1306では入力画像はフレームメモリAに蓄えられる。なお、ここで述べたフレームメモリAとは、ソフトウェア符号化器のメモリ領域（例えば、第4図のメモリ405内にこのメモリ領域が確保される）の一部を意味している。ステップ1307では、現在符号化中の
- 5 フレームがIフレームであるか否かが判定される。そして、Iフレームではない場合にはステップ1308で動き推定・動き補償処理が行われる。このステップ1308における処理の詳細を表すフローチャートの例を第14図に示す。まず、ステップ1401でフレームメモリAとB（フレームメモリBには前フレームの復号画像が格納されている）に蓄
- 10 えられた画像の間でグローバル動き推定が行われ、グローバル動きパラメータとして、画像の隅の点の動きベクトルが出力バッファに出力される。ステップ1402では、この画像の隅の点の動きベクトルを用いて式(9)、(10)により代表点の動きベクトルが計算される。続いてステップ1403では、変数Mに0が代入される。Mは画像内のラインの
- 15 番号を表し、Mが0であることは、画像の最も上のラインを処理中であることを意味し、Mが画像のライン数から1を引いた値であるときには、画像の最も下のラインを処理中であることを意味する。ステップ1402で計算された代表点の動きベクトルを用いて、ステップ1404では式(11)により第Mラインの仮代表点の動きベクトルが計算される。そしてこの仮代表点の動きベクトルを活用してステップ1405では第M
- 20 ラインに含まれる画素全ての動きベクトルが式(12)により計算され、求められた動きベクトルに従って、フレームメモリBに格納されている前フレームの復号画像を用いてグローバル動き補償予測画像の第Mラインが合成され、フレームメモリFに蓄えられる。ステップ1406では
- 25 Mの値に1が加えられ、ステップ1407ではMの値が画像のライン数に等しければステップ1408へ、等しく無ければステップ1404に

移動する。ステップ 1 4 0 8 の処理が開始される時点では、フレームメモリ D には、グローバル動き補償による予測画像が蓄えられている。ステップ 1 4 0 8 以降では、ブロックマッチングの処理が行われる。まず

5 ステップ 1 4 0 8 では、フレームメモリ F とフレームメモリ A（入力画像）との間でブロックごとに動き推定の処理が行われ、各ブロックの動きベクトルが求められ、その動きベクトルは出力バッファに出力される。続いてこの動きベクトルと、フレームメモリ F に蓄えられた画像を用いてステップ 1 4 0 9 ではブロックマッチングによる予測画像が合成され、これが最終的な予測画像となってフレームメモリ C に蓄えられる。そして

10 ステップ 1 4 1 0 ではフレームメモリ A と C の差分画像が求められ、これがフレームメモリ A に蓄えられる。

ここで第 13 図に戻る。ステップ 1 3 0 8 における処理が開始される直前、フレームメモリ A には、現フレームが I フレームである場合には入力画像が、現フレームが P フレームである場合には入力画像と予測画像の差分画像が蓄えられている。ステップ 1 3 0 8 では、このフレーム

15 メモリ A に蓄えられた画像に対して D C T が適用され、ここで計算された D C T 係数は量子化された後に出力バッファに出力される。さらにステップ 1 3 1 0 で、この量子化 D C T 係数には逆量子化され、逆 D C T が適用され、この結果得られた画像はフレームメモリ B に格納される。

20 続いてステップ 1 3 1 1 では、再び現フレームが I フレームであるか否かが判定され、I フレームではない場合にはステップ 1 3 1 2 でフレームメモリ B と C の画像が加算され、この結果がフレームメモリ B に格納される。ここで、1 フレーム分の符号化処理が終了することになる。そして、ステップ 1 3 1 3 の処理が行われる直前にフレームメモリ B に格

25 納されている画像は、符号化処理が終了したばかりのフレームの再生画像（復号側で得られるものと同じ）である。ステップ 1 3 1 3 では、符

号化が終了したフレームが最後のフレームであるか否かが判定され、最後のフレームであれば、符号化処理が終了する。最後のフレームではない場合には、ステップ 1 3 1 4 で N に 1 が加算され、再びステップ 1 3 0 3 に戻って次のフレームの符号化処理が開始される。なお、ここで説明したフローチャートはグローバル動き補償を行なった結果合成されたグローバル動き補償予測画像に対してブロックマッチングを適用する方法（第 8 図の動き補償処理部 8 0 1 を使用する装置に対応する方法）に関するものであるが、グローバル動き補償とブロックマッチングを並列に行う方法（第 9 図の動き補償処理部 9 0 1 を使用する装置に対応する方法）に関するフローチャートもわずかな変更を加えるのみで作成できることは明らかである。

一方、ソフトウェア復号化装置 5 0 0 では、入力された符号化ビットストリーム 5 0 1 は、一旦入力バッファ 5 0 2 に蓄えられた後に、汎用プロセッサ 5 0 3 に読み込まれる。汎用プロセッサ 5 0 3 はハードディスクやフロッピーディスクなどによる蓄積デバイス 5 0 8 から読み出されたプログラムを蓄えるプログラム用メモリ 5 0 4 および処理用メモリ 5 0 5 を活用して復号化処理を行う。この結果得られた復号化画像は、一旦出力フレームメモリ 5 0 6 に蓄えられた後に、出力画像 5 0 7 として出力される。

第 15 図は、第 5 図に示したソフトウェア復号化装置上で動作する復号化ソフトウェアのフローチャートを示す。1 5 0 1 で処理が開始され、まずステップ 1 5 0 2 で入力情報があるか否かが判定される。ここで入力情報が無ければステップ 1 5 0 3 で復号化の処理を終了する。入力情報がある場合には、まず、ステップ 1 5 0 4 でフレームタイプ情報が入力される。なお、この「入力される」とは、入力バッファ 5 0 2 に蓄えられた情報を読み込むことを意味している。ステップ 1 5 0 5 では、読

み込んだフレームタイプ情報が' I ' であるか否かが判定される。そして、' I ' ではない場合には、ステップ 1 5 0 6 で予測画像合成処理が行われる。このステップ 1 5 0 6 で行われる処理の詳細をフローチャートを第 16 図に示す。

- 5 まず、ステップ 16 0 1 で画像の隅の点の動きベクトルが入力される。ステップ 16 0 2 では、この画像の隅の点の動きベクトルを用いて式(9)、(10)により代表点の動きベクトルが計算される。続いてステップ 16 0 3 では、変数Mに0が代入される。Mは画像内のラインの番号を表し、Mが0であることは、画像の最も上のラインを処理中である
- 10 ことを意味し、Mが画像のライン数から1を引いた値であるときには、画像の最も下のラインを処理中であることを意味する。ステップ 16 0 2 で計算された代表点の動きベクトルを用いて、ステップ 16 0 4 では式(11)により第Mラインの仮代表点の動きベクトルが計算される。そしてこの仮代表点の動きベクトルを活用してステップ 16 0 5 では第M
- 15 ラインに含まれる画素すべての動きベクトルが式(12)により計算され、求められた動きベクトルに従って、フレームメモリ E に格納されている前フレームの復号画像を用いてグローバル動き補償予測画像の第Mラインが合成され、フレームメモリ G に蓄えられる。なお、ここで述べたフレームメモリ G とは、ソフトウェア復号化器のメモリ 5 0 5 の領域の一部
- 20 を意味している。ステップ 16 0 6 ではMの値に1が加えられ、ステップ 16 0 7 ではMの値が画像のライン数に等しければステップ 16 0 8 へ、等しく無ければステップ 16 0 4 に移動する。ステップ 16 0 8 の処理が開始される時点では、フレームメモリ G には、グローバル動き補償による予測画像が蓄えられている。ステップ 16 0 8 では、ブロックマ
- 25 ッチングの処理が行われる。ブロックごとの動きベクトル情報が入力され、この動きベクトルとフレームメモリ G に格納された画像を用いてブ

ロックマッチングによる予測画像が合成され、この予測画像はフレームメモリDに格納される。

- ここで第 15 図に戻る。ステップ 1 5 0 7 では量子化 D C T 係数が入力され、これに逆量子化、逆 D C T を適用して得られた画像がフレーム
- 5 メモリ E に格納される。ステップ 1 5 0 8 では、再び現在復号化中のフレームが I フレームであるか否かが判定される。そして、I フレームではない場合には、ステップ 1 5 0 9 でフレームメモリ D と E に格納された画像が加算され、この結果の画像がフレームメモリ E に格納される。ステップ 1 5 1 0 の処理を行う直前にフレームメモリ E に格納されている
- 10 画像が、再生画像となる。ステップ 1 5 1 0 では、このフレームメモリ E に格納された画像が出力フレームメモリ 5 0 6 に出力され、そのまま出力画像として復号化器から出力される。こうして 1 フレーム分の復号化処理が終了し、処理は再びステップ 1 5 0 2 に戻る。

- 第 4 図と第 5 図に示したソフトウェア画像符号化装置、ソフトウェア
- 15 画像復号化装置に本明細書で示したフレーム間予測画像の合成方法を実行するプログラムを実行させると、グローバル動き補償やワーピング予測の処理をより少ない演算量で実現することが可能となる。このため、本発明を用いない場合と比較して、消費電力の低減、装置の低価格化、より大きな画像を実時間で処理できるようになる、画像符号化・復号化
- 20 以外の処理を含む同時並列処理を行うことが可能となる、等の効果を期待することができる。また、本明細書で示したアルゴリズムを用いることにより、従来の画像復号化装置では演算能力の限界から実時間で再生できなかったような圧縮画像データを、実時間で再生することが可能となる。

- 25 以上本発明の実施形態について述べたが、以下のような実施形態も本発明に含まれる。

- (1) 従来型の画像符号化方法では、フレーム間予測を行った後に離散コサイン変換などによる誤差符号化が行われるが、フレーム間予測画像をそのまま再生画像として使用する画像符号化方法・復号化方法に対しても、本発明は有効である。
- 5 (2) 本明細書では、画像の形状は長方形であることを仮定したが、長方形以外の任意の形状を持つ画像にも、本発明は適用可能である。この場合、まず任意形状の画像を囲む長方形に対して本発明の処理を適用し、任意形状画像に含まれる画素に対してのみ動きベクトルを求める演算を行えば良い。
- 10 (3) 本明細書では、 p 又は q の値が 2 の負ではない整数乗であることを前提として 2 段階の処理による動きベクトルの内・外挿アルゴリズムを示した。しかし、 p 及び q が 2 の負ではない整数乗ではない場合でも、この 2 段階処理アルゴリズムは除算における分母の値を小さくするという効果を持っており、レジスタのオーバーフローを防ぐ意味で有効である。
- 15

産業上の利用可能性

- 図 17 に、本明細書で示した予測画像合成方法を用いる符号化・復号化装置の具体例を示す。(a) は、パソコン 1701 に画像符号化・復号化用のソフトウェアを組み込むことにより、画像符号化・復号化装置として活用する場合を示す。このソフトウェアは何らかの蓄積メディア (CD-ROM、フロッピーディスク、ハードディスクなど) に記録されており、これをパソコンが読み込んで使用する。また、さらに何らかの通信回線にこのパソコンを接続することにより、映像通信端末として
- 20
- 25 活用することも可能となる。

(b) は本発明による符号化方法による動画像情報を蓄積メディア 1

702に記録した符号化ビットストリームを読み取り、本発明による装置を持つ再生装置1703で再生し、再生された映像信号をテレビモニタ1704に表示する場合を示す。再生装置1703は符号化ビットストリームを読み取るだけであり、テレビモニタ1704内に復号化装置
5 が組み込まれている場合もある。

(c)は、ディジタル放送用のテレビ受信機1705に本発明の復号化装置を組み込んだ場合を示す。また、(d)は、ケーブルテレビ用のケーブル1708又は衛星/地上波放送のアンテナに接続されたセットトップボックス1709内に復号化装置を実装し、これをテレビモニタ17
10 10で再生する場合を示す。このとき、(b)の1704と同様に、セットトップボックスではなく、テレビモニタ内に符号化装置を組み込んでも良い。

(e)は、ディジタル携帯端末1706に本発明の符号化器、復号化器を組み込んだ場合を示す。ディジタル携帯端末の場合、符号器・復号
15 化器を両方持つ送受信型の端末の他に、符号化器のみの送信端末、復号化器のみの受信端末の3通りの実装形式のいずれでもよい。

(f)は、動画像撮影用のカメラ1707の中に符号化装置を組み込む場合を示す。また、カメラ1707は映像信号を取り込むのみであり、これを専用の符号化装置1711に組み込む構成でもよい。この図に示
20 したいずれの装置・システムに関しても、本明細書に示した方法を実装することにより、従来の技術を活用した場合と比較して、装置を簡略化することが可能となる。

請求の範囲

1. 第1のフレーム画像と時間的に異なる第2のフレーム画像から第1のフレーム画像の予測画像を合成する方法において、

第1のフレーム画像と第2のフレーム画像から第1のフレーム画像の座標 (i, j) 、 $(i + p, j)$ 、 $(i, j + q)$ 、 $(i + p, j + q)$ に (i, j, p, q) は整数) を持つ4つの代表点の動きベクトル動きベクトル(動きベクトルの水平・垂直成分が $1/k$ の整数倍の値をとり(ただし、 k は2の $h k$ 乗、かつ $h k$ は負ではない整数) を求める第1ステップと、

10 画素のサンプリング間隔を水平、垂直方向共に1として、サンプリング点が座の水平、垂直成分が、共に整数に w を加えた数である点の上に存在している画像を対象として(ただし、 $w = w_n / w_d$ 、かつ w_n は負ではない整数、かつ w_d は2の $h w$ 乗、かつ $h w$ は負ではない整数、かつ $w_n < w_d$)、4個の代表点における動きベクトルに対し、共1次

15 内・外挿を行うことによって座標 $(x + w, y + w)$ に位置する画素の動きベクトルを計算する第2ステップを有し、

上記第2ステップが、座標 (i, j) と $(i, j + q)$ に位置する代表点の動きベクトルに対して線形内・外挿を行うことにより、座標 $(i, y + w)$ に位置する点の動きベクトルの水平・垂直成分をそれぞれ $1/z$ の整数倍をとる数値として(ただし、 z は2の $h z$ 乗、かつ $h z$ は負ではない整数) 求め、さらに座標 $(i + p, j)$ と $(i + p, j + q)$ に位置する代表点の動きベクトルに対して

20 線形内・外挿を行うことにより、座標 $(i + p, y + w)$ に位置する点の動きベクトルの水平・垂直成分をそれぞれ $1/z$ の整数倍をとる

数値として求める第3ステップと、

その後、座標 $(i, y+w)$ と $(i+p, y+w)$ に位置する上記2個の動きベクトルに対して線形内・外挿を行うことにより、座標 $(x+w, y+w)$ に位置する画素の動きベクトルの水平・垂直成分をそれぞれ $1/m$ の整数倍をとる数値として（ただし、 m は2の h_m 乗、かつ h_m は負ではない整数）求める第4ステップ途をもつことを特徴とするフレーム間予測画像の合成方法。

2. 第1のフレーム画像と時間的に異なる第2のフレーム画像から第1のフレーム画像の予測画像を合成する方法において、

- 10 第1のフレーム画像と第2のフレーム画像から第1のフレーム画像の座標 (i, j) 、 $(i+p, j)$ 、 $(i, j+q)$ 、 $(i+p, j+q)$ に (i, j, p, q) は整数）を持つ4つの代表点の動きベクトル動きベクトル（動きベクトルの水平・垂直成分が $1/k$ の整数倍の値をとり（ただし、 k は2の h_k 乗、かつ h_k は負ではない整数）を求め
- 15 る第1ステップと、

画素のサンプリング間隔を水平、垂直方向共に1として、サンプリング点が座標の水平、垂直成分が、共に整数に w を加えた数である点の上に存在している画像を対象として（ただし、 $w = w_n/w_d$ 、かつ w_n は負ではない整数、かつ w_d は2の h_w 乗、かつ h_w は負ではない整数、かつ $w_n < w_d$ ）、4個の代表点における動きベクトルに対し、共

20 1次内・外挿を行うことによって座標 $(x+w, y+w)$ に位置する画素の動きベクトルを計算する第2ステップを有し、

上記第2ステップが、座標 (i, j) と $(i+p, j)$ に位置する代表点の動きベクトルに対して線形内・外挿を行うことにより、座標

- ($x + w$, j) に位置する点の動きベクトルの水平・垂直成分をそれぞれ $1/z$ の整数倍をとる数値として (ただし、 z は 2 の h_z 乗、かつ h_z は負ではない整数) 求め、さらに座標 (i , $j + q$) と ($i + p$, $j + q$) に位置する代表点の動きベクトルに対して線形内・外挿を行うことにより、座標 ($x + w$, $j + q$) に位置する点の動きベクトルの水平・垂直成分をそれぞれ $1/z$ の整数倍をとる数値として求める第 3 ステップと、

- その後、座標 ($x + w$, j) と ($x + w$, $j + p$) に位置する上記 2 個の動きベクトルに対して線形内・外挿を行うことにより、座標 ($x + w$, $y + w$) に位置する画素の動きベクトルの水平・垂直成分をそれぞれ $1/m$ の整数倍をとる数値として (ただし、 m は 2 の h_m 乗、かつ h_m は負ではない整数) 求める第 4 ステップとをもつことを特徴とするフレーム間予測画像の合成方法。

3. 上記座標 (i , j)、($i + p$, j)、(i , $j + q$)、($i + p$, $j + q$) に位置する代表点の動きベクトルの水平・垂直成分を k 倍したものである (u_0 , v_0)、(u_1 , v_1)、(u_2 , v_2)、(u_3 , v_3) を用いて座標 ($x + w$, $y + w$) に位置する画素の動きベクトルを求めるときに、座標 (i , $y + w$) に位置する点の動きベクトルの水平・垂直成分をそれぞれ z 倍したものである ($u_L(y + w)$, $v_L(y + w)$) を、

$$u_L(y + w) = ((q \cdot w_d - y \cdot w_d - w_n) u_0 + (y \cdot w_d + w_n) u_2) z) // (q \cdot k \cdot w_d),$$

$$v_L(y + w) = ((q \cdot w_d - y \cdot w_d - w_n) v_0 + (y \cdot w_d + w_n) v_2) z) // (q \cdot k \cdot w_d)$$

を計算することにより（ただし、「////」は通常を除算による演算結果が整数ではない場合にこれを近隣の整数に丸め込む除算で、演算子としての優先順位は乗除算と同等）求め、

更に座標 $(i + p, y + w)$ に位置する点の動きベクトルの水平・

- 5 垂直成分をそれぞれ z 倍したものである $(u_R(y + w), v_R(y + w))$ を、

$$u_R(y + w) = ((q \cdot w_d - y \cdot w_d - w_n) u_1 + (y \cdot w_d + w_n) u_3) z) //// (q \cdot k \cdot w_d),$$

$$v_R(y + w) = ((q \cdot w_d - y \cdot w_d - w_n) v_1 + (y \cdot w_d + w_n) v_3) z) //// (q \cdot k \cdot w_d),$$

10

を計算することにより求めた後に、座標 $(x + w, y + w)$ に位置する画素の動きベクトルの水平・垂直成分をそれぞれ m 倍したものである $(u(x + w, y + w), v(x + w, y + w))$ を

$$u(x + w, y + w) = ((p \cdot w_d - x \cdot w_d - w_n) u_L(y + w) + (x \cdot w_d + w_n) u_R(y + w)) m) // (p \cdot z \cdot w_d)$$

15

$$v(x + w, y + w) = ((p \cdot w_d - x \cdot w_d - w_n) v_L(y + w) + (x \cdot w_d + w_n) v_R(y + w)) m) // (p \cdot z \cdot w_d)$$

を計算することによって（ただし、「//」は通常を除算による演算結果が整数ではない場合にこれを近隣の整数に丸め込む除算で、演算子としての優先順位は乗除算と同等）求めることを特徴とする請求項 1

20

に記載のフレーム間予測画像の合成方法。

4. 座標 (i, j) 、 $(i + p, j)$ 、 $(i, j + q)$ 、 $(i + p, j + q)$ に位置する代表点の動きベクトルの水平・垂直成分を k 倍したものである (u_0, v_0) 、 (u_1, v_1) 、 (u_2, v_2) 、 $(u_3, v$

3) を用いて座標 $(x + w, y + w)$ に位置する画素の動きベクトルを求めるときに、座標 $(x + w, j)$ に位置する点の動きベクトルの水平・垂直成分をそれぞれ z 倍したものである $(uT(x + w), vT(x + w))$ を、

$$5 \quad uT(x + w) = (((p \cdot wd - x \cdot wd - wn) u0 + (x \cdot wd + wn) u1) z) // (p \cdot k \cdot wd) ,$$

$$vT(x + w) = (((p \cdot wd - x \cdot wd - wn) v0 + (x \cdot wd + wn) v1) z) // (p \cdot k \cdot wd) ,$$

を計算することにより（ただし、「////」は通常を除算による演算結果が整数ではない場合にこれを近隣の整数に丸め込む除算で、演算子としての優先順位は乗除算と同等）求め、

さらに座標 $(x + w, j + p)$ に位置する点の動きベクトルの水平・垂直成分をそれぞれ z 倍したものである $(uB(y + w), vB(y + w))$ を、

$$15 \quad uB(x + w) = (((p \cdot wd - x \cdot wd - wn) u2 + (x \cdot wd + wn) u3) z) // (p \cdot k \cdot wd) ,$$

$$vB(x + w) = (((p \cdot wd - x \cdot wd - wn) v2 + (x \cdot wd + wn) v3) z) // (p \cdot k \cdot wd) ,$$

を計算することにより求めた後に、座標 $(x + w, y + w)$ に位置する画素の動きベクトルの水平・垂直成分をそれぞれ m 倍したものである $(u(x + w, y + w), v(x + w, y + w))$ を

$$20 \quad u(x + w, y + w) = (((q \cdot wd - y \cdot wd - wn) uT(x + w) + (y \cdot wd + wn) uB(x + w)) m) // (q \cdot z \cdot wd) ,$$

$$v(x + w, y + w) = (((q \cdot wd - y \cdot wd - wn) vT(x + w) +$$

$$(y \cdot wd + wn) \vee B(x + w) \cdot m) // (q \cdot z \cdot wd),$$

を計算することによって（ただし、「//」は通常の除算による演算結果が整数ではない場合にこれを近隣の整数に丸め込む除算で、演算子としての優先順位は乗除算と同等）求めることを特徴とする請求項 2

5 に記載のフレーム間予測画像の合成方法。

5. p の絶対値が 2 の α 乗（ α は負ではない整数）であることを特徴とする請求項 1 に記載のフレーム間予測画像の合成方法。

6. q の絶対値が 2 の β 乗（ β は負ではない整数）であることを特徴とする請求項 2 又は 4 に記載のフレーム間予測画像の合成方法。

10 7. p と q の絶対値がそれぞれ 2 の α 乗と β 乗（ α 、 β は負ではない整数）であることを特徴とする請求項 1 に記載のフレーム間予測画像の合成方法。

8. p と q の絶対値がそれぞれ 2 の α 乗と β 乗（ α 、 β は負ではない整数）であることを特徴とする請求項 2 に記載のフレーム間予測画

15 像の合成方法。

9. $\alpha + hz$ が 8 の正の整数倍であり、かつ w が 0 であることを特徴とする請求項 5 に記載のフレーム間予測画像の合成方法。

10. $\beta + hz$ が 8 の正の整数倍であり、かつ w が 0 であることを特徴とする請求項 6 に記載のフレーム間予測画像の合成方法。

20 11. $\alpha + hz + hw$ が 8 の正の整数倍であり、かつ $w > 0$ であることを特徴とする請求項 5 に記載のフレーム間予測画像の合成方法。

12. $\beta + hz + hw$ が 8 の正の整数倍であり、かつ $w > 0$ であることを特徴とする請求項 6 に記載のフレーム間予測画像の合成方法。

13. 複数の異なる α の値に対応し、 $\alpha + hz$ が 16 以下となるよう

に hz の値を α の値に応じて変化させることを特徴とする請求項 9 に記載のフレーム間予測画像の合成方法。

14. 複数の異なる β の値に対応し、 $\beta + hz$ が 16 以下となるように hz の値を β の値に応じて変化させることを特徴とする請求項 10

5 に記載のフレーム間予測画像の合成方法。

15. 複数の異なる α の値に対応し、 $\alpha + hz + hw$ が 16 以下となるように hz の値を α の値に応じて変化させることを特徴とする請求項 11 に記載のフレーム間予測画像の合成方法。

10 16. 複数の異なる β の値に対応し、 $\beta + hz + hw$ が 16 以下となるように hz の値を β の値に応じて変化させることを特徴とする請求項 12 に記載のフレーム間予測画像の合成方法。

17. $z \geq m$ であることを特徴とする請求項 1 ないし 16 に記載のフレーム間予測画像の合成方法。

15 18. $k \geq z$ であることを特徴とする請求項 1 ないし 16 に記載のフレーム間予測画像の合成方法。

19. p と q の絶対値がそれぞれ画像の水平と垂直の画素数と異なることを特徴とする請求項 1 ないし 16 に記載のフレーム間予測画像の合成方法。

20 20. r を画像の水平方向の画素数、 s を画像の垂直方向の画素数として（ただし、 r と s は正の整数）、 p の絶対値を $1/2$ 倍した値は r より小さく、かつ p の絶対値は r 以上で、かつ q の絶対値を $1/2$ 倍した値は s より小さく、かつ q の絶対値は s 以上であることを特徴とする請求項 1 ないし 16 に記載のフレーム間予測画像の合成方法。

21. r を画像の水平方向の画素数、 s を画像の垂直方向の画素数と

し（ただし、 r と s は正の整数）で、 p の絶対値は r 以下であり、
 かつ p の絶対値を 2 倍した値は r より大きく、かつ q の絶対値は s 以下であり、かつ q の絶対値を 2 倍した値は s より大きいことを特徴とする請求項 1 ないし 16 に記載のフレーム間予測画像の合成方法。

- 5 22. 画像の水平方向と垂直方向の画素数がそれぞれ r と s であり（ただし、 r と s は正の整数）、かつ画像の画素が水平座標が 0 以上 r 未満、垂直座標が 0 以上 s 未満の範囲に存在しているときに、
- 座標 $(-c, -c)$ 、 $(r-c, -c)$ 、 $(-c, s-c)$ 、 $(r-c, s-c)$ に位置する画像の隅の点上に存在し（ただし、 $c = c_n$ / c_d 、かつ c_n は負ではない整数、かつ c_d は正の整数、かつ $c_n < c_d$ ）、
- 10 水平・垂直成分が $1/n$ の整数倍の値をとる動きベクトル（ただし、 n は正の整数）を n 倍したものである (u_{00}, v_{00}) 、 (u_{01}, v_{01}) 、 (u_{02}, v_{02}) 、 (u_{03}, v_{03}) （ただし、 u_{00} 、 v_{00} 、 u_{01} 、 v_{01} 、 u_{02} 、 v_{02} 、 u_{03} 、 v_{03} は整数）を用いて、
- 15 $u'(x, y) = ((s \cdot c_d - c_n - y \cdot c_d) ((r \cdot c_d - c_n - x \cdot c_d) u_{00} + (x \cdot c_d + c_n) u_{01}) + (y \cdot c_d + c_n) ((r \cdot c_d - c_n - x \cdot c_d) u_{02} + (x \cdot c_d + c_n) u_{03})) k) / ((r \cdot s \cdot n \cdot c_d))$,
- $v'(x, y) = ((s \cdot c_d - c_n - y \cdot c_d) ((r \cdot c_d - c_n - x \cdot c_d) v_{00} + (x \cdot c_d + c_n) v_{01}) + (y \cdot c_d + c_n) ((r \cdot c_d - c_n - x \cdot c_d) v_{02} + (x \cdot c_d + c_n) v_{03})) k) / ((r \cdot s \cdot n \cdot c_d \cdot c_d))$,
- 20 $u_0 = u'(i, j)$,
- $v_0 = v'(i, j)$,

$$u1 = u' (i + p, j),$$

$$v1 = v' (i + p, j),$$

$$u2 = u' (i, j + q),$$

$$v2 = v' (i, j + q),$$

$$5 \quad u3 = u' (i + p, j + q),$$

$$v3 = v' (i + p, j + q),$$

で表される $(u0, v0)$ 、 $(u1, v1)$ 、 $(u2, v2)$ 、 $(u3, v3)$ を（ただし、「 $///$ 」は通常を除算による演算結果が整数ではない場合にこれを近隣の整数に丸め込む除算で、演算子としての優先順位は乗除算と同等）、代表点 (i, j) 、 $(i + p, j)$ 、 $(i, j + q)$ 、 $(i + p, j + q)$ の動きベクトルの水平・垂直成分を k 倍したものとして使用することを特徴とする請求項 1 ないし 2 1 に記載のフレーム間予測画像の合成方法。

2 3. 符号化しようとする現フレームの画像信号とフレーム間予測
15 画像との差を誤差画像としてを出力する第 1 ステップと

上記誤差画像を信号変換し、変換信号を得てそれを符号化する第 2 ステップと

上記変換信号を逆変換して上記誤差画像の復号誤差画像を作る第 4 ステップと

20 上記復号誤差画像と上記フレーム間予測画像を用いて上記現フレームの画像信号の次の現フレームの画像に対応するフレーム間予測画像信号を作る第 5 ステップとを有し、

上記第 5 ステップが請求項 1 ないし 1 6 のいずれかのフレーム間予測画像の合成方法で行うことを特徴とする画像符号化方法。

24. 上記第5ステップが代表点の動きベクトルに関する情報を検出し符号化するステップを含む請求項23記載の画像符号化方法。

25. 上記第5ステップが代表点が画像の隅の点である請求項23に記載の画像の符号化方法。

5 26. 復号すべき画像フレームのフレーム間符号化信号と上記画像フレームの動きベクトルの情報とを入力する第1ステップと、

上記フレーム間符号化信号を復号誤差信号に変換する第2ステップと、

10 上記復号すべき画像フレームと時間的に異なる他の画像フレームの復号画像信号と上記動きベクトルの情報からフレーム間予測画像を作る第3ステップと、

復号誤差信号と上記フレーム間予測画像の信号とを加算して、上記復号すべき画像フレームの復号画像信号を得る第4ステップともち、

15 上記第3ステップが請求項1ないし16のいずれかのフレーム間予測画像の合成方法で行う画像復号化方法。

27. 上記複数の代表点が符号化データとして直接符号化されている代表点の動きベクトルに関する情報を再生して用いる上記画像の隅の点である請求項26に記載の画像の復号化方法。

20 28. 上記代表点が上記画像の隅の点である請求項26に記載の画像の復号化方法。

29. 符号化しようとする現フレームの画像信号とフレーム間予測像と像を信号変換する第1変換部の出力の一部を符号化する符号化器と、

第1変換部の出力の一部を逆変換して上記誤差画像の復号誤差画像

を得る第2変換部と、

上記復号誤差画像と上記フレーム間予測画像とから上記現フレームの画像信号の復号画像信号を得る復号手段と、

上記前フレームの復号画像と現フレームの入力画像が入力され前述
5 のフレーム間予測画像の合成を行う動き補償処理部をもつ符号化装置であって、

動き補償処理部が上記前フレームの復号画像と現フレームの入力画像から上記前フレームの復号画像の座標 (i, j) 、 $(i + p, j)$ 、 $(i, j + q)$ 、 $(i + p, j + q)$ に (i, j, p, q) は整数) を
10 持つ4つの代表点の動きベクトル（動きベクトルの水平・垂直成分が $1/k$ の整数倍の値をとり（ただし、 k は2の $h k$ 乗、かつ $h k$ は負ではない整数）を求めるグローバル動きベクトル推定部と、

上記動きベクトルと上記前フレームの復号画像から上記符号化しようとする現フレームの画像の信号を予測したフレーム間予測画像を作る予測画像合成部と持ち、
15

上記予測画像合成部が、画素のサンプリング間隔を水平、垂直方向共に1として、サンプリング点が座標の水平、垂直成分が、共に整数に w を加えた数である点の上に存在している画像を対象として（ただし、 $w = w_n / w_d$ 、かつ w_n は負ではない整数、かつ w_d は2の $h w$ 乗、
20 かつ $h w$ は負ではない整数、かつ $w_n < w_d$ ）、座標 (i, j) と $(i, j + q)$ に位置する代表点の動きベクトルに対して線形内・外挿を行うことにより、座標 $(i, j + w)$ に位置する点の動きベクトルの水平・垂直成分をそれぞれ $1/z$ の整数倍をとる数値として（ただし、 z は2の $h z$ 乗、かつ $h z$ は負ではない整数）求め、さらに座標 $(i +$

- p, j) と $(i + p, j + q)$ に位置する代表点の動きベクトルに対して線形内・外挿を行うことにより、座標 $(i + p, y + w)$ に位置する点の動きベクトルの水平・垂直成分をそれぞれ $1/z$ の整数倍をとる数値として求めた後に、 $(i, y + w)$ と $(i + p, y + w)$ に
- 5 位置する上記2個の動きベクトルに対して線形内・外挿を行うことにより、座標 $(x + w, y + w)$ に位置する画素の動きベクトルの水平・垂直成分をそれぞれ $1/m$ の整数倍をとる数値として（ただし、 m は2の h_m 乗、かつ h_m は負ではない整数）求める演算部と、
- 上記座標 $(x + w, y + w)$ に位置する画素の動きベクトルと上記前
- 10 フレームの復号画像から予測画像を合成する合成部とをもつ画像符号化装置。

30. 符号化しようとする現フレームの画像信号とフレーム間予測画像信号との差を誤差画像としてを出力する減算器と、

- 上記誤差画像を信号変換する第1変換部の出力の一部を符号化する
- 15 符号化器と、

第1変換部の出力の一部を逆変換して上記誤差画像の復号誤差画像を得る第2変換部と、

上記復号誤差画像と上記フレーム間予測画像とから上記現フレームの画像信号の復号画像信号を得る復号手段と、

- 20 上記前フレームの復号画像と現フレームの入力画像601が入力され上記フレーム間予測画像の合成を行う動き補償処理部をもつ符号化装置であって、

動き補償処理部が上記前フレームの復号画像と現フレームの入力画像から上記前フレームの復号画像の座標 (i, j) 、 $(i + p, j)$ 、

($i, j + q$)、($i + p, j + q$)に(i, j, p, q は整数)を持つ4つの代表点の動きベクトル(動きベクトルの水平・垂直成分が $1/k$ の整数倍の値をとり(ただし、 k は2の $h k$ 乗、かつ $h k$ は負ではない整数)を求めるグローバル動きベクトル推定部と、

- 5 上記動きベクトルと上記前フレームの復号画像から上記符号化しようとする現フレームの画像の信号を予測したフレーム間予測画像を作る予測画像合成部と持ち、

- 上記予測画像合成部が、画素のサンプリング間隔を水平、垂直方向共に1として、サンプリング点が座標の水平、垂直成分が、共に整数
- 10 に w を加えた数である点の上に存在している画像を対象として(ただし、 $w = w_n / w_d$ 、かつ w_n は負ではない整数、かつ w_d は2の $h w$ 乗、かつ $h w$ は負ではない整数、かつ $w_n < w_d$)、座標(i, j)と($i + p, j$)に位置する代表点の動きベクトルに対して線形内・外挿を行うことにより、座標($x + w, j$)に位置する点の動きベクトルの水
- 15 平・垂直成分をそれぞれ $1/z$ の整数倍をとる数値として(ただし、 z は2の $h z$ 乗、かつ $h z$ は負ではない整数)求め、

- さらに座標($i, j + q$)と($i + p, j + q$)に位置する代表点の動きベクトルに対して線形内・外挿を行うことにより、座標($x + w, j + q$)に位置する点の動きベクトルの水平・垂直成分をそれぞれ
- 20 れ $1/z$ の整数倍をとる数値として求めた後に、

($x + w, j$)と($x + w, j + p$)に位置する上記2個の動きベクトルに対して線形内・外挿を行うことにより、座標($x + w, y + w$)に位置する画素の動きベクトルの水平・垂直成分をそれぞれ $1/m$ の整数倍をとる数値として(ただし、 m は2の $h m$ 乗、かつ $h m$ は負ではな

い整数) 求める演算部と、

上記 $(x + w, y + w)$ に位置する画素の動きベクトルと上記前フレームの復号画像から予測画像を合成する合成部とをもつ画像符号化装置。

- 5 31. 座標 (i, j) 、 $(i + p, j)$ 、 $(i, j + q)$ 、 $(i + p, j + q)$ に位置する代表点の動きベクトルの水平・垂直成分を k 倍したものである (u_0, v_0) 、 (u_1, v_1) 、 (u_2, v_2) 、 (u_3, v_3) を用いて座標 $(x + w, y + w)$ に位置する画素の動きベクトルを求めるときに、座標 $(i, y + w)$ に位置する点の動きベクトルの
- 10 水平・垂直成分をそれぞれ z 倍したものである $(u_L(y + w), v_L(y + w))$ を、

$$u_L(y + w) = ((q \cdot w_d - y \cdot w_d - w_n) u_0 + (y \cdot w_d + w_n) u_2) z) // (q \cdot k \cdot w_d),$$

$$v_L(y + w) = ((q \cdot w_d - y \cdot w_d - w_n) v_0 + (y \cdot w_d + w_n)$$

- 15 $v_2) z) // (q \cdot k \cdot w_d)$

を計算することにより (ただし、「 $////$ 」は通常の除算による演算結果が整数ではない場合にこれを近隣の整数に丸め込む除算で、演算子としての優先順位は乗除算と同等) 求め、

- 更に座標 $(i + p, y + w)$ に位置する点の動きベクトルの水平・
- 20 垂直成分をそれぞれ z 倍したものである $(u_R(y + w), v_R(y + w))$ を、

$$u_R(y + w) = ((q \cdot w_d - y \cdot w_d - w_n) u_1 + (y \cdot w_d + w_n) u_3) z) // (q \cdot k \cdot w_d),$$

$$v_R(y + w) = ((q \cdot w_d - y \cdot w_d - w_n) v_1 + (y \cdot w_d + w_n)$$

$v_3) z) // (q \cdot k \cdot wd)$ 、

を計算することにより求めた後に、座標 $(x + w, y + w)$ に位置する画素の動きベクトルの水平・垂直成分をそれぞれ m 倍したものである $(u(x + w, y + w), v(x + w, y + w))$ を

$$5 \quad u(x + w, y + w) = ((p \cdot wd - x \cdot wd - wn) u_L(y + w) + (x \cdot wd + wn) u_R(y + w)) m // (p \cdot z \cdot wd)$$

$$v(x + w, y + w) = ((p \cdot wd - x \cdot wd - wn) v_L(y + w) + (x \cdot wd + wn) v_R(y + w)) m // (p \cdot z \cdot wd)$$

を計算することによって（ただし、「//」は通常を除算による演算結

- 10 果が整数ではない場合にこれを近隣の整数に丸め込む除算で、演算子としての優先順位は乗除算と同等）求めることを特徴とする請求項 29 に記載の符号化装置。

32. 座標 (i, j) 、 $(i + p, j)$ 、 $(i, j + q)$ 、 $(i + p, j + q)$ に位置する代表点の動きベクトルの水平・垂直成分を k

- 15 倍したものである (u_0, v_0) 、 (u_1, v_1) 、 (u_2, v_2) 、 (u_3, v_3) を用いて座標 $(x + w, y + w)$ に位置する画素の動きベクトルを求めるときに、

座標 $(x + w, j)$ に位置する点の動きベクトルの水平・垂直成分をそれぞれ z 倍したものである $(u_T(x + w), v_T(x + w))$ を、

$$20 \quad u_T(x + w) = ((p \cdot wd - x \cdot wd - wn) u_0 + (x \cdot wd + wn) u_1) z // (p \cdot k \cdot wd)$$

$$v_T(x + w) = ((p \cdot wd - x \cdot wd - wn) v_0 + (x \cdot wd + wn) v_1) z // (p \cdot k \cdot wd)$$

を計算することにより（ただし、「////」は通常を除算による演算結

果が整数ではない場合にこれを近隣の整数に丸め込む除算で、演算子としての優先順位は乗除算と同等) 求め、

さらに座標 $(x + w, j + p)$ に位置する点の動きベクトルの水平・垂直成分をそれぞれ z 倍したものである $(u_B(y + w), v_B(y + w))$

5 を、

$$u_B(x + w) = ((p \cdot w_d - x \cdot w_d - w_n) u_2 + (x \cdot w_d + w_n) u_3) z) // (p \cdot k \cdot w_d),$$

$$v_B(x + w) = ((p \cdot w_d - x \cdot w_d - w_n) v_2 + (x \cdot w_d + w_n) v_3) z) // (p \cdot k \cdot w_d),$$

10 を計算することにより求めた後に、座標 $(x + w, y + w)$ に位置する画素の動きベクトルの水平・垂直成分をそれぞれ m 倍したものである $(u(x + w, y + w), v(x + w, y + w))$ を

$$u(x + w, y + w) = ((q \cdot w_d - y \cdot w_d - w_n) u_T(x + w) + (y \cdot w_d + w_n) u_B(x + w)) m) // (q \cdot z \cdot w_d),$$

15 $v(x + w, y + w) = ((q \cdot w_d - y \cdot w_d - w_n) v_T(x + w) + (y \cdot w_d + w_n) v_B(x + w)) m) // (q \cdot z \cdot w_d),$

を計算することによって(ただし、「//」は通常を除算による演算結果が整数ではない場合にこれを近隣の整数に丸め込む除算で、演算子としての優先順位は乗除算と同等) 求めること請求項 30 に記載のフ

20 レーム間予測画像の符号化装置。

33. p の絶対値が 2 の α 乗 (α は負ではない整数) であるとする請求項 29 に記載のフレーム間予測画像の符号化装置。

34. q の絶対値が 2 の β 乗 (β は負ではない整数) である請求項 30 に記載のフレーム間予測画像の符号化装置。

35. p と q の絶対値がそれぞれ 2 の α 乗と β 乗 (α 、 β は負ではない整数) である請求項 29 に記載の符号化装置。

36. p と q の絶対値がそれぞれ 2 の α 乗と β 乗 (α 、 β は負ではない整数) である請求項 30 に記載の符号化装置。

5 37. $\alpha + hz$ が 8 の正の整数倍であり、かつ w が 0 である請求項 33 に記載の符号化装置。

38. $\beta + hz$ が 8 の正の整数倍であり、かつ w が 0 である請求項 34 に記載の符号化装置。

39. $\alpha + hz + hw$ が 8 の正の整数倍であり、かつ $w > 0$ である
10 請求項 33 に記載の符号化装置。

40. $\beta + hz + hw$ が 8 の正の整数倍であり、かつ $w > 0$ である請求項 34 に記載の符号化装置。

41. 複数の異なる α の値に対応し、 $\alpha + hz$ が 16 以下となるように hz の値を α の値に応じて変化する請求項 37 に記載の符号化装置。

15 42. 複数の異なる β の値に対応し、 $\beta + hz$ が 16 以下となるように hz の値を β の値に応じて変化する請求項 38 に記載の符号化装置。

43. 複数の異なる α の値に対応し、 $\alpha + hz + hw$ が 16 以下となるように hz の値を α の値に応じて変化する請求項 39 に記載の符号化装置。

20 44. 複数の異なる β の値に対応し、 $\beta + hz + hw$ が 16 以下となるように hz の値を β の値に応じて変化する請求項 40 に記載の符号化装置。

45. 上記動き補償処理部が更に上記代表点の動きベクトルに関する情報を符号化する手段をもつ請求項 29 ないし 40 のいずれか記載

の符号化装置。

46. 上記代表点が画像の隅の位置の点である請求項29ないし40のいずれかに記載の符号化装置。

47. 上記第1変換部及び第2変換部がそれぞれ上記誤差誤差画像の信号をDCT変換して量子化する回路及び逆量子化して逆DCT変換する回路である請求項29ないし40のいずれかに記載の符号化装置。

48. 符号化された画像信号のフレーム間誤差の符号を誤差画像の信号に変換する変換回路と、復号されたフレーム画像信号を記憶するフレームメモリと、上記符号化された画像信号の動きベクトルと上記フレームメモリの復号されたフレーム画像信号を入力し、予測画像を合成する予測画像合成部と、上記予測画像合成部の出力と上記変換回路の出力を加算して復号画像を作る加算部と、上記加算部の出力を上記フレームメモリに記録する手段を持つ画像復号化装置において、上記予測画像合成部が請求項1ないし16のいずれかに記載された記載されたフレーム間予測画像の合成方法を行う手段で構成された画像復号化装置。

49. 請求項1ないし22に記載のフレーム間予測画像の合成方法を実行するためのソフトウェアを記録した蓄積メディア。

50. 請求項48に記載の画像復号化装置を駆動するためのソフトウェアを記録した蓄積メディア。

51. 請求項23、24、又は25に記載の符号化方法によつ生成された符号化されたビットストリームを記録した蓄積メディア。

52. 請求項26、27又は28に記載の画像復号化方法によって復号化することができる符号化された圧縮ビットストリームを記録した

蓄積メディア。

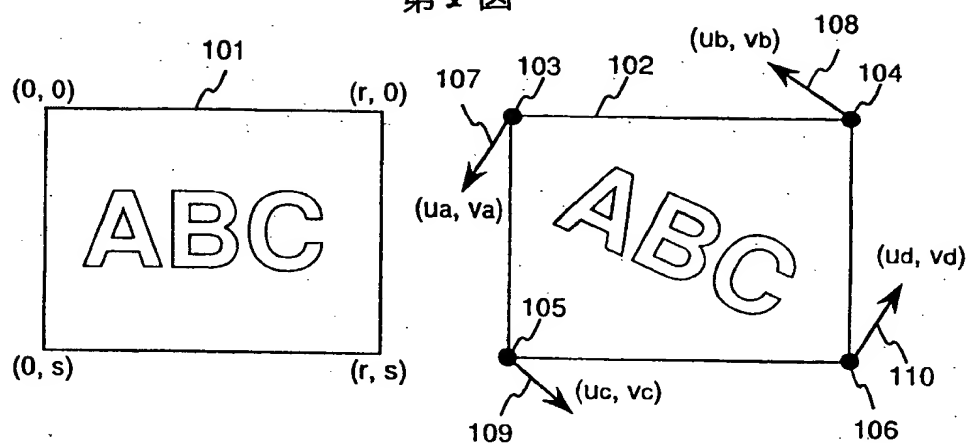
5

10

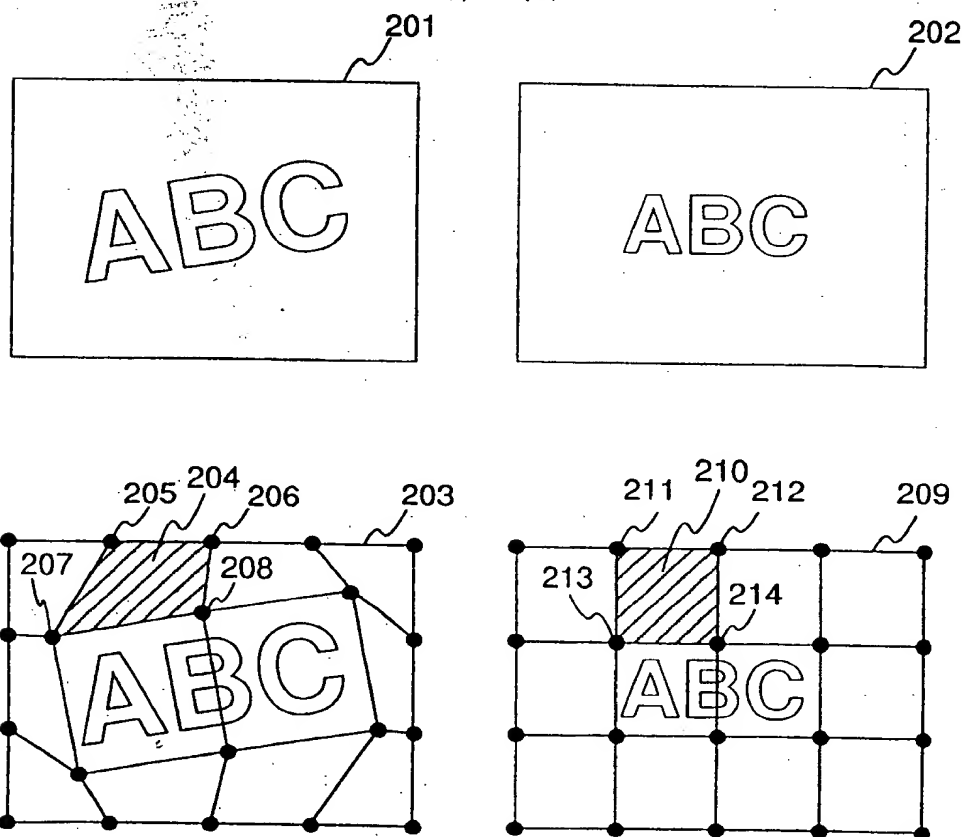
15

This Page Blank (uspto)

第 1 図

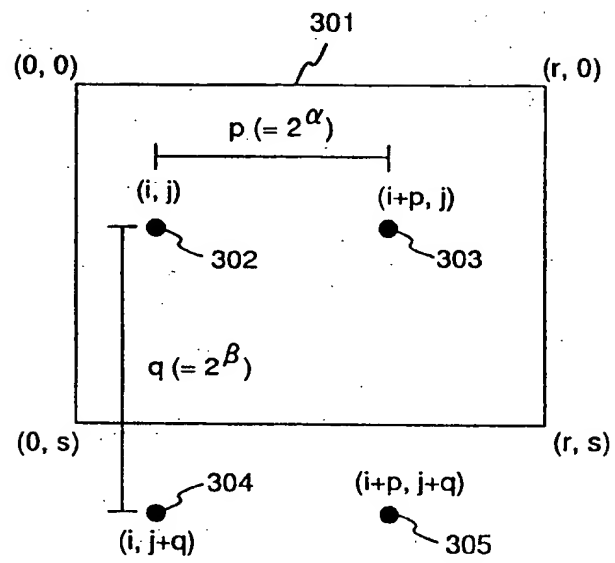


第 2 図



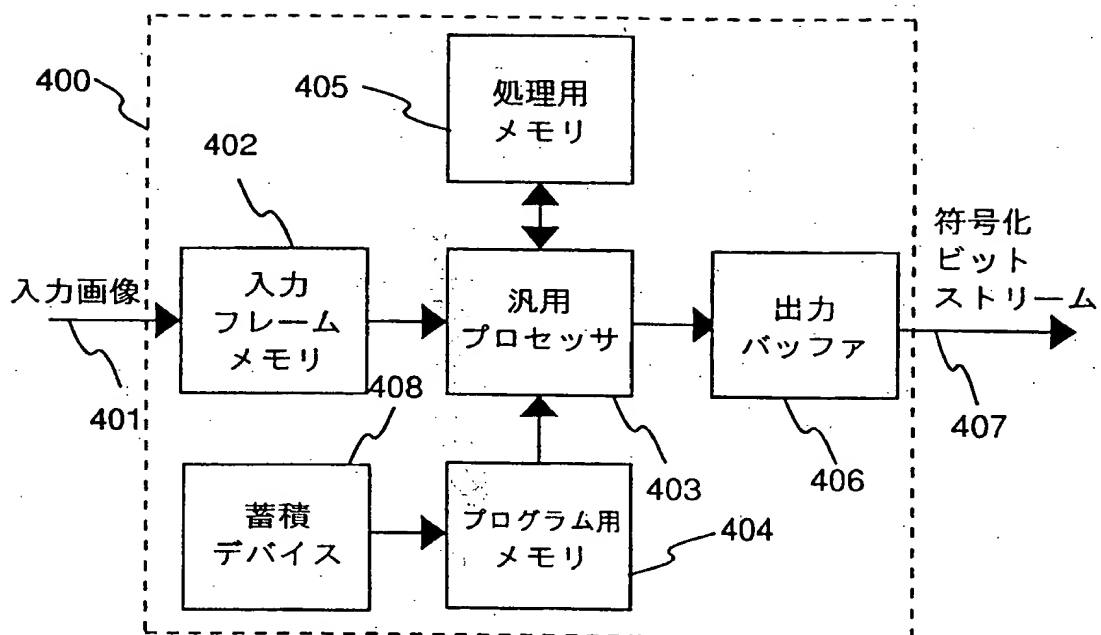
This Page Blank (uspto)

第 3 図

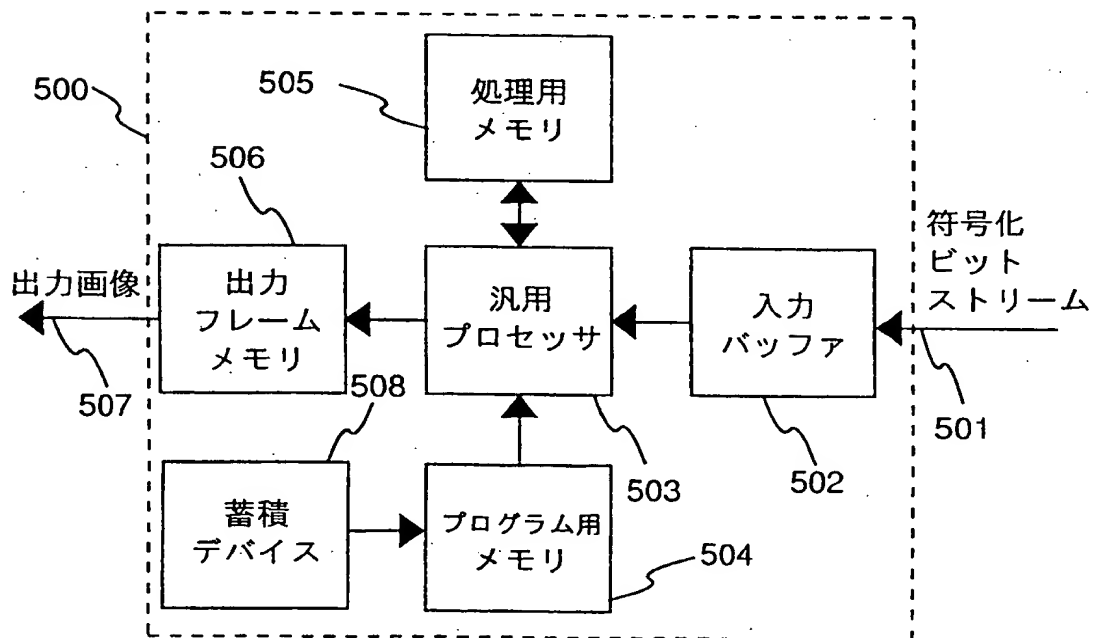


This Page Blank (uspto)

第4図

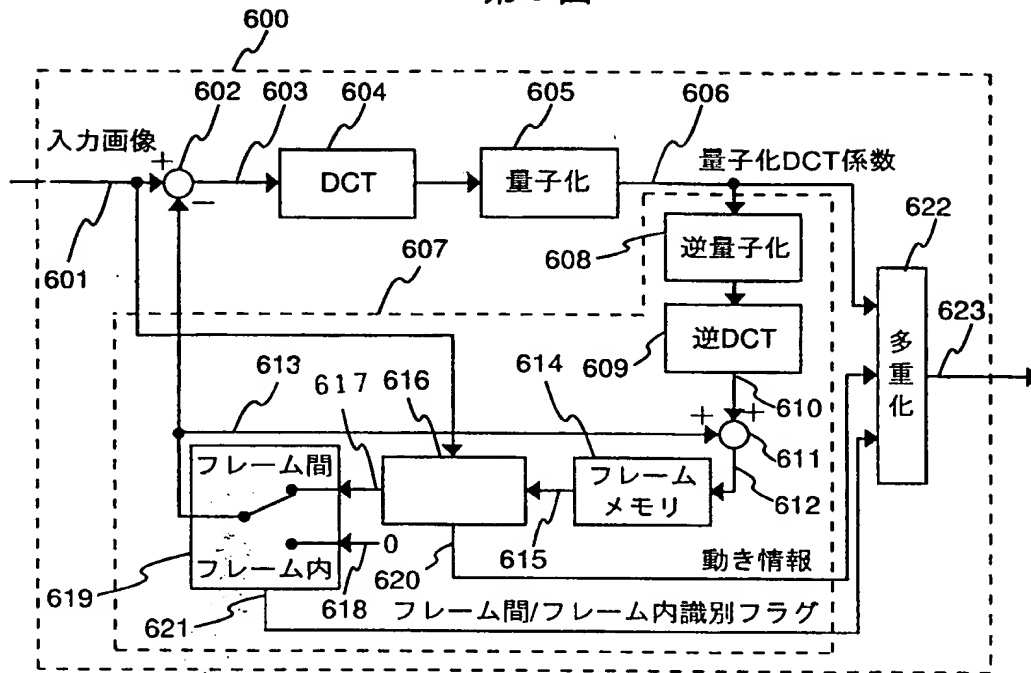


第5図

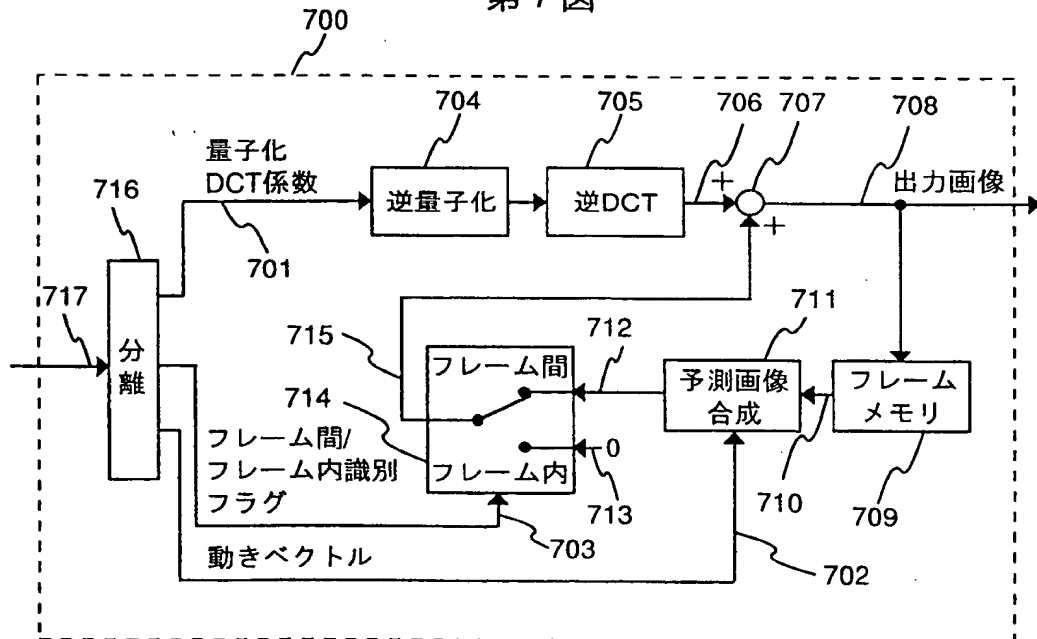


This Page Blank (uspto)

第 6 図

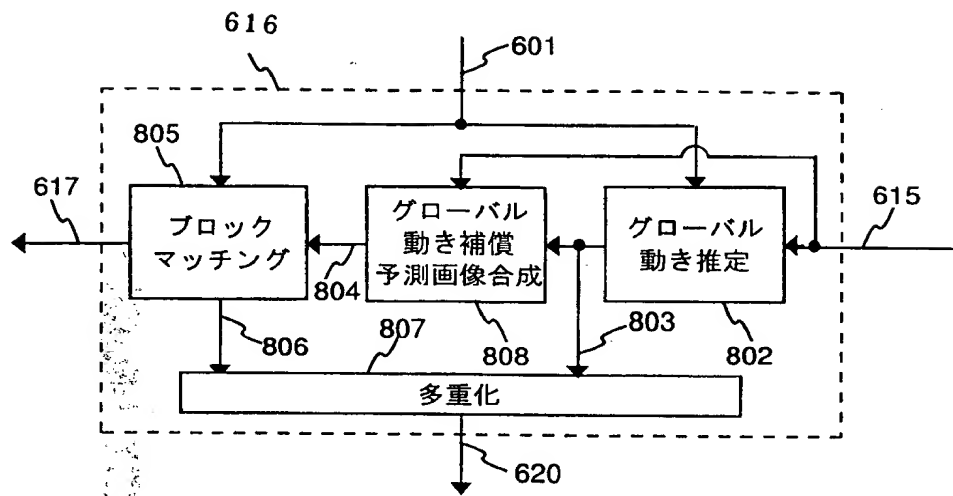


第 7 圖

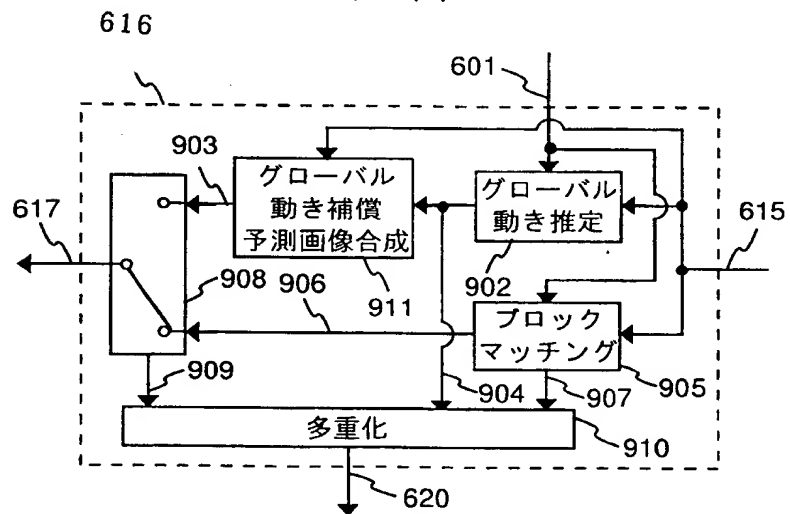


This Page Blank (uspto)

第 8 図

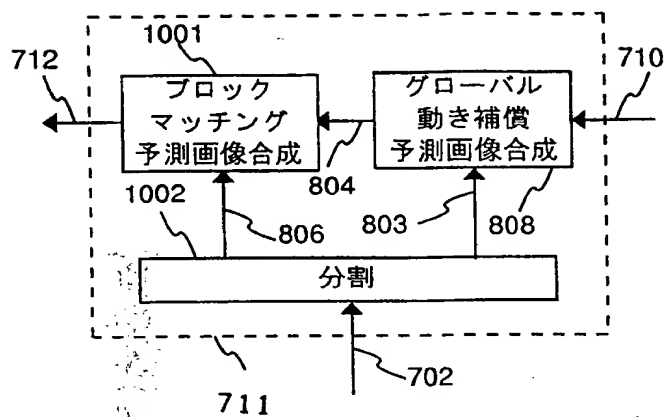


第 9 図

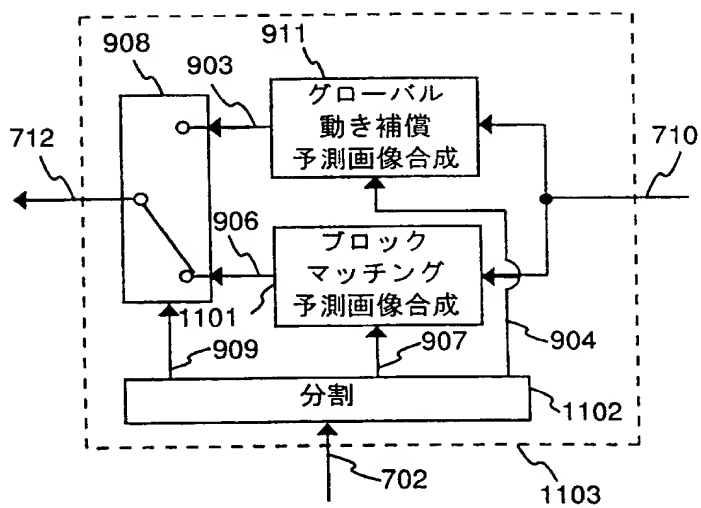


This Page Blank (uspto)

第10図

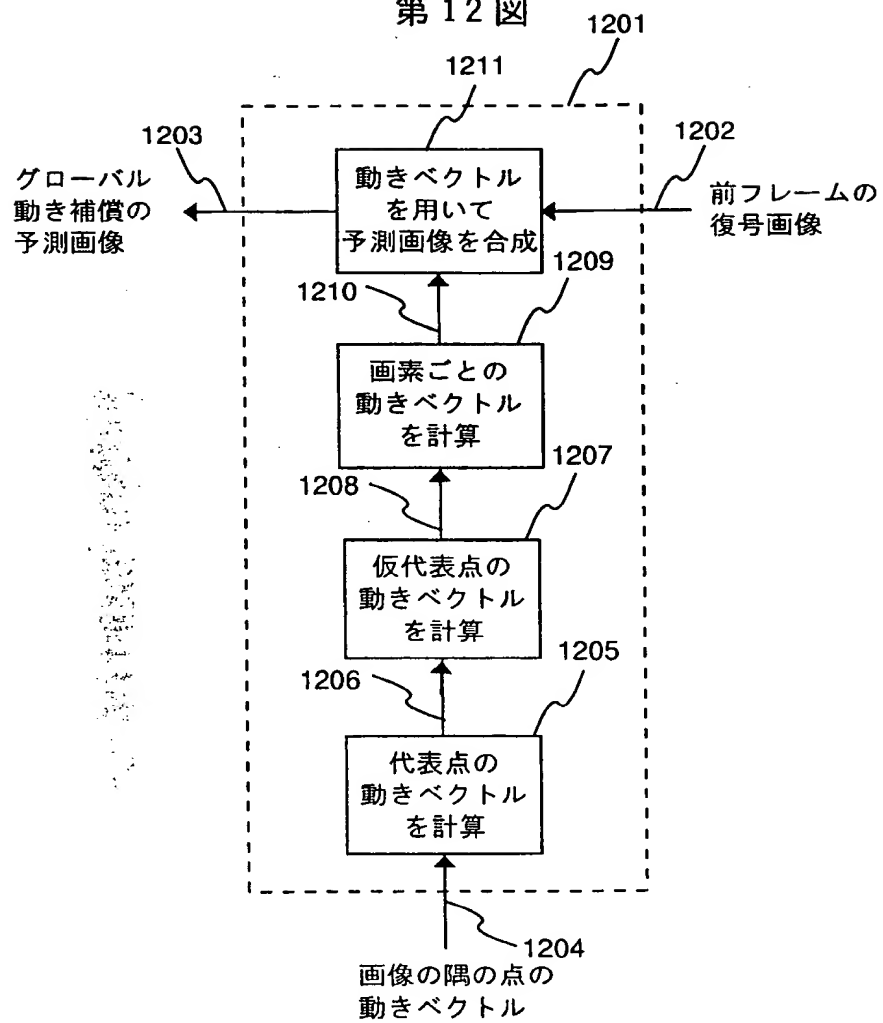


第11図



This Page Blank (uspto)

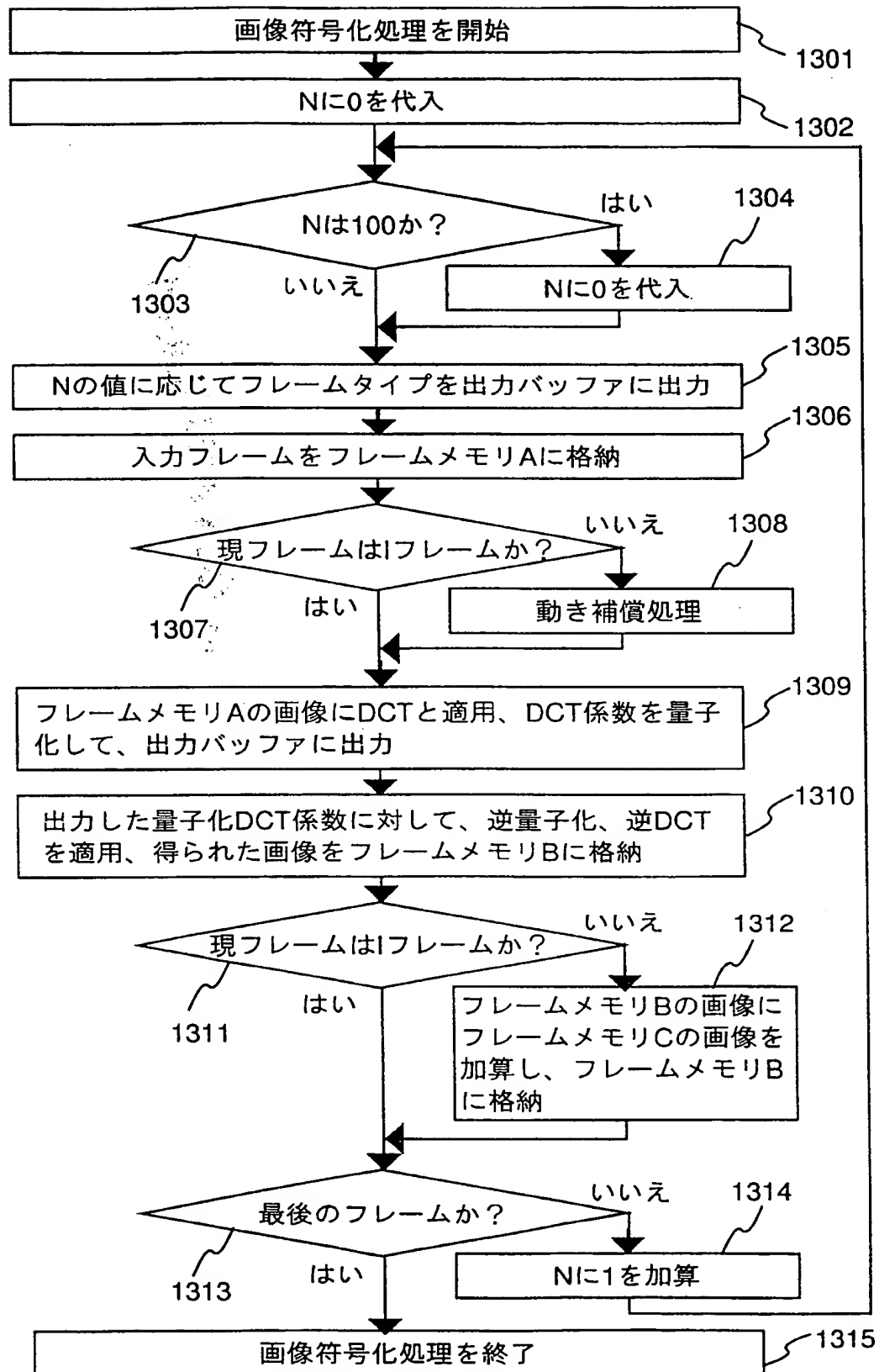
第 12 図



This Page Blank (uspto)

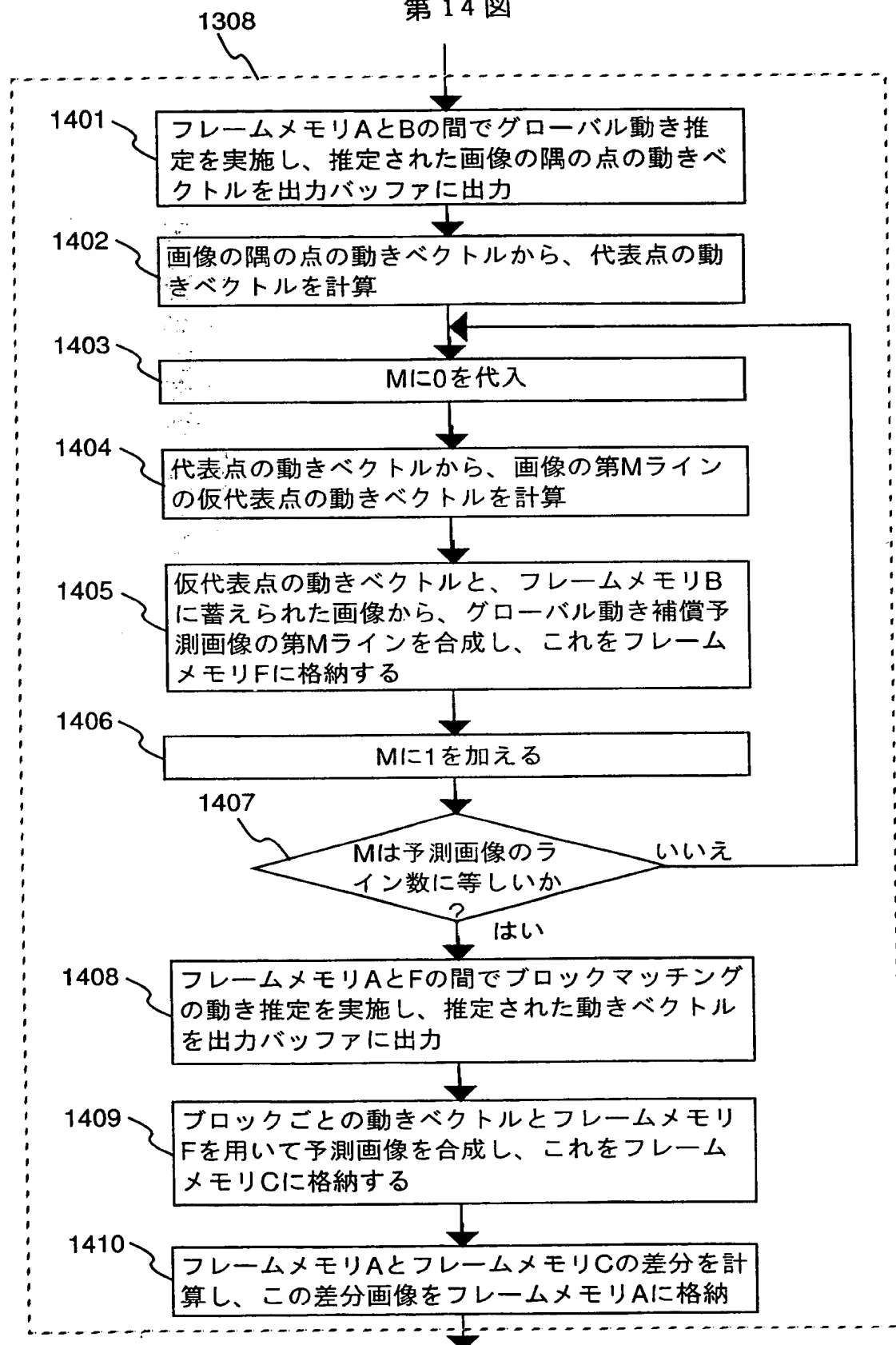
8 / 12

第 13 図



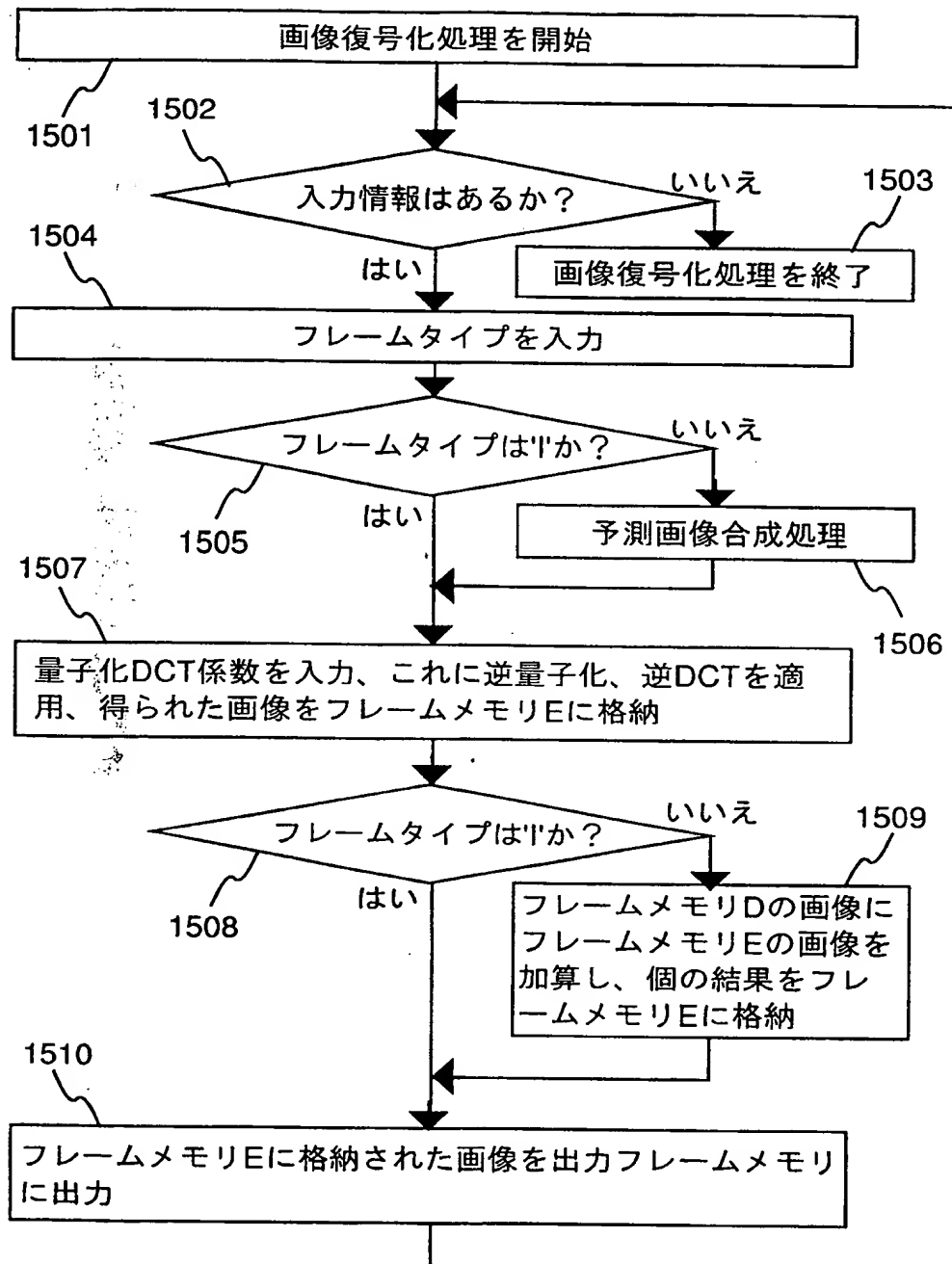
This Page Blank (uspto)

第 14 図



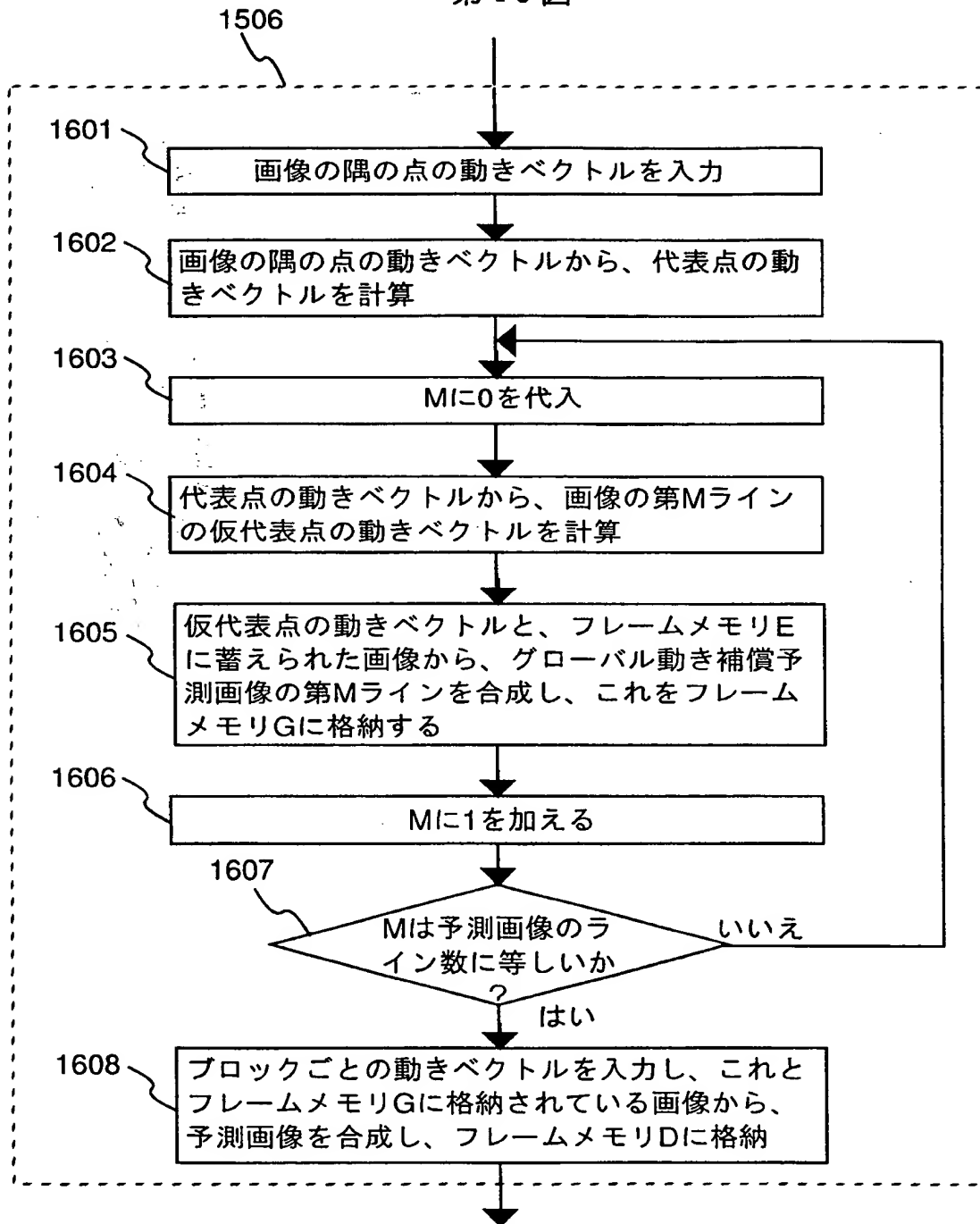
This Page Blank (uspto)

第 15 図



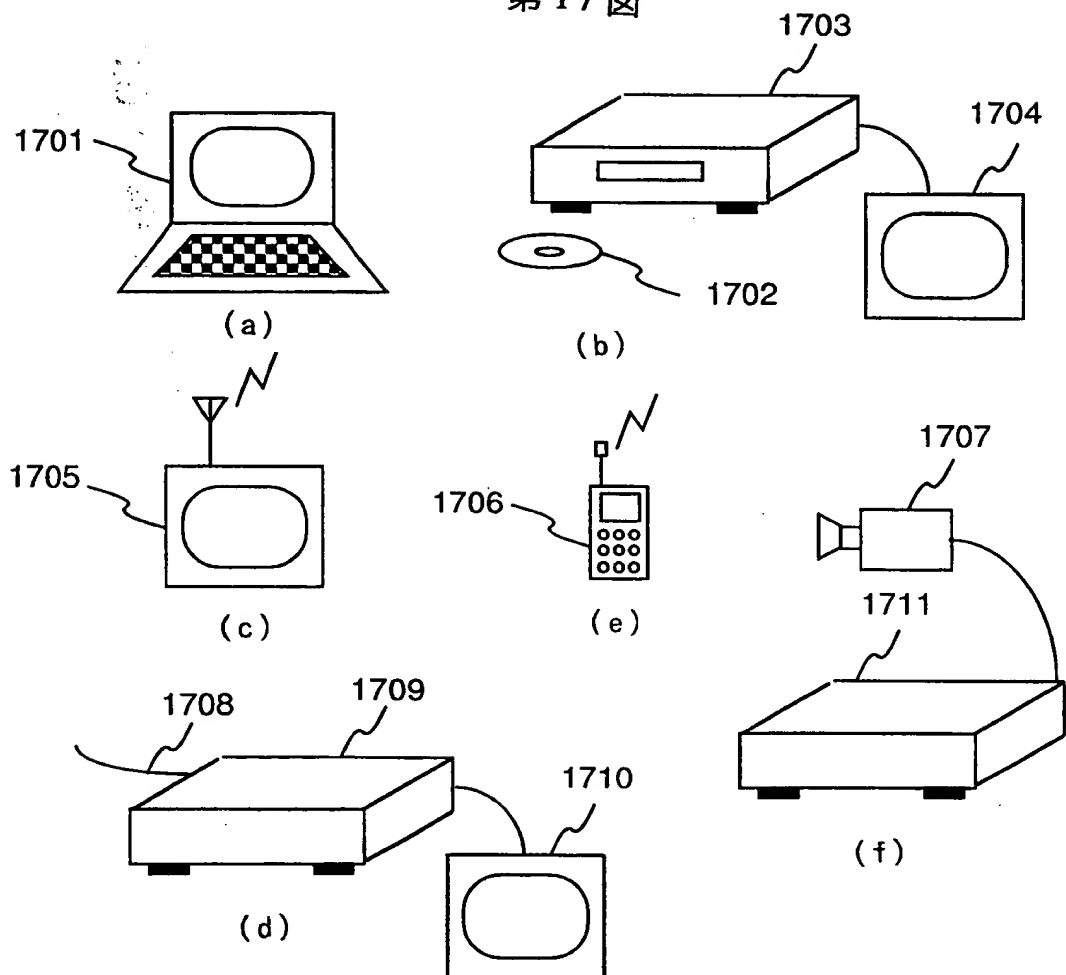
This Page Blank (uspto)

第 16 図



This Page Blank (uspto)

第 17 図



This Page Blank (uspto)

INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP98/02435

A. CLASSIFICATION OF SUBJECT MATTER
Int.Cl.⁶ H04N7/32

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
Int.Cl.⁶ H04N7/24-7/68Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
Jitsuyo Shinan Koho 1957-1998
Kokai Jitsuyo Shinan Koho 1975-1998

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	C. Haung, et al., "A New Motion Compensation Method for Image Sequence Coding Using Hierarchical Grid Interpolation" IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, Vol. 4, No. 1 (1994), p.42-52	1-52
A	Seiji Kimura et al., "Motion Compensation Based on Variable Size Variable Shape Block (in Japanese)", Technical Report of the Institute of Television Engineers of Japan, Vol. 20, No. 64 (1996) (Tokyo) p.31-38	1-52

☐ Further documents are listed in the continuation of Box C.
 ☐ See patent family annex.

* Special categories of cited documents:	"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A" document defining the general state of the art which is not considered to be of particular relevance	"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"E" earlier document but published on or after the international filing date	"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"&" document member of the same patent family
"O" document referring to an oral disclosure, use, exhibition or other means	
"P" document published prior to the international filing date but later than the priority date claimed	

Date of the actual completion of the international search
25 August, 1998 (25. 08. 98)Date of mailing of the international search report
8 September, 1998 (08. 09. 98)Name and mailing address of the ISA/
Japanese Patent Office

Authorized officer

Facsimile No.

Telephone No.

This Page Blank (uspto)

国際調査報告

国際出願番号 PCT/J P 98/02435

A. 発明の属する分野の分類 (国際特許分類 (IPC))

Int. Cl⁶ H04N7/32

B. 調査を行った分野

調査を行った最小限資料 (国際特許分類 (IPC))

Int. Cl⁶ H04N7/24-7/68

最小限資料以外の資料で調査を行った分野に含まれるもの

日本国実用新案公報 1957-1998年
日本国公開実用新案公報 1975-1998年

国際調査で使用した電子データベース (データベースの名称、調査に使用した用語)

C. 関連すると認められる文献

引用文献の カテゴリー*	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求の範囲の番号
A	C. Haung, et al 「A New Motion Compensation Method for Image Sequence Coding Using Hierarchical Grid Interpolation」 IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, 第4巻, 第1号 (1994), p. 42-52	1-52
A	木村青司, 他 「可変サイズ可変形状ブロックに基づいた動き補償法」 テレビジョン学会技術報告, 第20巻, 第64号 (1996) (東京) p. 31-38	1-52

☐ C欄の続きにも文献が列挙されている。☐ パテントファミリーに関する別紙を参照。

* 引用文献のカテゴリー

- 「A」 特に関連のある文献ではなく、一般的技術水準を示すもの
「E」 先行文献ではあるが、国際出願日以後に公表されたもの
「L」 優先権主張に疑義を提起する文献又は他の文献の発行日若しくは他の特別な理由を確立するために引用する文献 (理由を付す)
「O」 口頭による開示、使用、展示等に言及する文献
「P」 国際出願日前で、かつ優先権の主張の基礎となる出願

の日の後に公表された文献

- 「T」 国際出願日又は優先日後に公表された文献であって出願と矛盾するものではなく、発明の原理又は理論の理解のために引用するもの
「X」 特に関連のある文献であって、当該文献のみで発明の新規性又は進歩性がないと考えられるもの
「Y」 特に関連のある文献であって、当該文献と他の1以上の文献との、当業者にとって自明である組合せによって進歩性がないと考えられるもの
「&」 同一パテントファミリー文献

国際調査を完了した日

25.08.98

国際調査報告の発送日

08.09.98

国際調査機関の名称及びあて先

日本国特許庁 (ISA/J P)

郵便番号100-8915

東京都千代田区霞が関三丁目4番3号

特許庁審査官 (権限のある職員)

國分 直樹

5C

9070

電話番号 03-3581-1101 内線 3542

This Page Blank (uspto)